ONLINE AND OFFLINE ENSEMBLE STRATEGIES FOR NOISE-AWARE UNSUPERVISED ANOMALOUS SOUND DETECTION

Technical Report

*Carl-Emil F. Krag, Christian Rhod, Emil Rosenlund, Simon Bøgh Bræck**

Aalborg University
cfk22, crhod22, erosen22, sc22 - @student.aau.dk

ABSTRACT

This technical report details an ensemble pipeline submitted for the DCASE 2026 Challenge Task 2: Noise-Aware Unsupervised Anomalous Sound Detection for Machine Condition Monitoring. Our approach fine-tunes BEATs and SSLAM with an auxiliary classification objective where each combination of machine type, domain, attributes, and spatial audio channel is treated as a distinct class. This encourages the models to capture machine-specific acoustic characteristics while exploiting channel-dependent spatial information. For anomaly detection, the extracted embeddings are processed by multiple anomaly scorers, including KNN, GMMs, Autoencoders, and local density normalized KNN. The outputs from these individual detectors across the fine-tuned BEATs and SSLAM models are aggregated using a min-pooling score-based fusion strategy. To address both offline and real-time inference requirements, we employed rank normalization to stabilize scores in the offline setting, and calibrated Z-score normalization to maintain test sample independence in the online setting. The proposed ensemble substantially outperforms the baseline system under both offline (+10.69) and online (+8.67) evaluation protocols.

***Index Terms*—** Anomalous sound detection, ensemble fusion, score normalization, multi-channel, BEATs, SSLAM,

1. INTRODUCTION

The Fourth Industrial Revolution is driving a digital transformation in manufacturing, where advanced acoustic monitoring has emerged as a cost-effective alternative to traditional vibration or thermal sensors for predictive maintenance. By continuously analyzing machine acoustics, anomalous sound detection (ASD) systems can detect structural deviations and mechanical failures, significantly reducing costly production downtime [1],[2].

The DCASE 2026 Task 2 challenge [3] advances this field by focusing on "Noise-Aware Unsupervised Anomalous Sound Detection," where systems must detect anomalies in new machine types without hyperparameter tuning, all while maintaining robustness against domain shifts such as variations in environmental noise or sensor placement. A critical shift in the 2026 edition is the introduction of two-channel audio recordings, which provide distinct spatial cues that are essential for decoupling localized machine signals from diffuse factory background noise.

To address these challenges, this project presents a high-performance pipeline that moves beyond standard reconstruction-based autoencoders[1]. We leverage the discriminative capacity of pre-trained audio transformers, specifically the bidirectional encoder representation from audio transformers (BEATs) [4] and the self-supervised light-weight audio model (SSLAM) [5]. By framing the ASD problem as an auxiliary classification task, we map input acoustic signals into a high-dimensional latent manifold. Unlike traditional approaches that identify anomalies based solely on local density deviations, our system employs an ensemble of heterogeneous scoring backends. This ensures that anomalous samples are identified through the consensus of diverse statistical methods, including proximity-based and reconstruction-based detectors, thereby increasing the robustness of the detection and reducing reliance on the failure modes of any single estimator. A sketch of the pipeline can be seen in Figure 1.

2. SUPERVISED FEATURE EXTRACTION

Traditional unsupervised models often suffer from "representational collapse," where the model learns to reconstruct the noise floor rather than the mechanical signal [6]. We treat feature extraction as a supervised task to ensure the latent manifold is anchored in the machine's true operational state.

2.1. Auxiliary Classification Fine-Tuning

Rather than using backbones as fixed feature extractors, we fine-tune them using an auxiliary classification objective. Labels are constructed by concatenating machine type, domain, attributes, and spatial channel IDs, creating 90 unique class labels, for instance: "ToyCarEmu_sec00_source_car_B1_spd_2_ch0".

By including the specific channel ID (ch0 or ch1) in the classification label, we force the transformer backbones to move beyond simple spectral analysis. Because the two microphones are placed at different physical distances from the target machine, the model must learn to distinguish between the direct-path mechanical sounds and the secondary reflections. Specifically, the attention mechanism is emphasized to minimize classification loss by detecting the slight time-of-arrival differences and intensity variations characteristic of each microphone's unique spatial perspective. This effectively trains the network to utilize the residual and differential signals between channels as discriminative features, allowing it to "spatially filter" ambient factory noise from the localized mechanical signal.

*Thanks to our supervisor Kevin Wilkinghoff for inspiration and feedback

[1]Source code: https://github.com/AAU-CE-8/DCASE_2026_AAU_CE_8_822

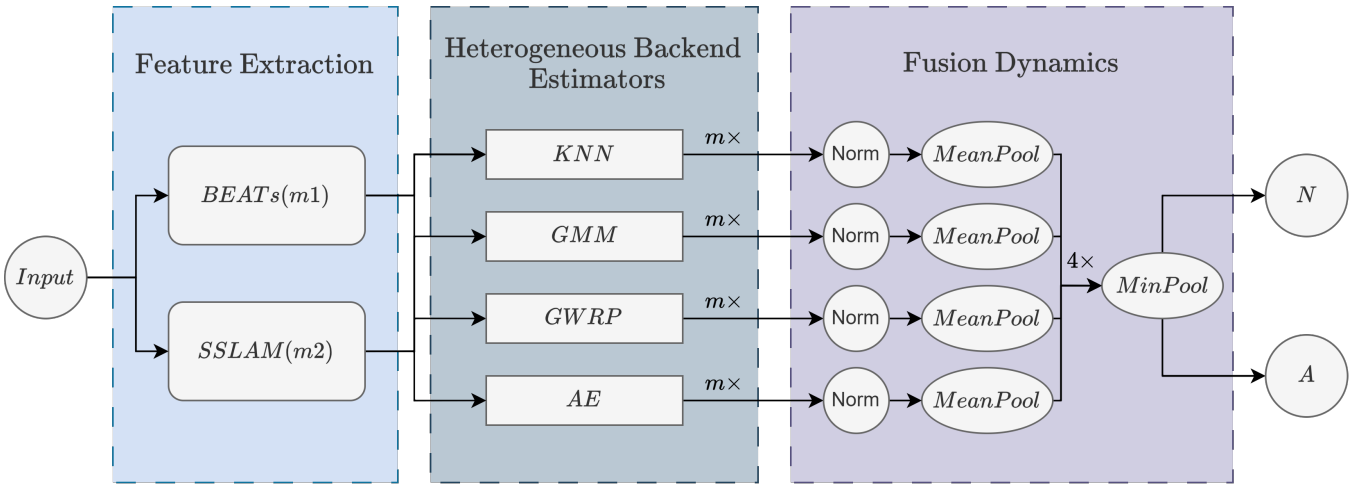

Figure 1: Overview of the proposed offline pipeline. Feature extractors generate embeddings from multiple backbone models (e.g., BEATs and SSLAM), which are processed by heterogeneous backend estimators. In the Fusion Dynamics stage, estimator outputs are normalized and mean-pooled across backbone models to obtain a single score per estimator. The estimator scores are subsequently fused using min pooling to generate the final anomaly score.

Through extensive experimentation, we found that unfreezing only the final 4 layers of the BEATs architecture preserves pretrained semantic knowledge, while SSLAM benefited from unfreezing the final 10 layers for optimal task adaptation. This targeted unfreezing allows the model to retain its generalized audio understanding while specializing in the unique structural sound signatures of the industrial machines provided in the DCASE 2026 dataset. By explicitly training for channel identification, we force the network to develop an internal representation of the sound propagation path, which is highly indicative of machine health.

2.2. ArcFace Angular Margin Loss

To enforce extreme separation within the embedding space, we employ the ArcFace (Additive Angular Margin Loss) objective [7]. Conventional Softmax loss often fails to produce highly separable clusters in high-dimensional embedding spaces, leading to overlapping regions for similar but distinct machine operating modes. ArcFace mitigates this by adding a margin m to the angular space, compelling the model to produce compact, well-separated clusters for each machine state. This minimizes intra-class variance and ensures that anomalies - which lack defined class centroids - are pushed toward the peripheries of the embedding manifold. The optimization process minimizes:

$$L_{Arc} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s(\cos(\theta_{y_i}+m))}}{e^{s(\cos(\theta_{y_i}+m))}+\sum_{j\neq y_i}e^{s\cos(\theta_j)}} \quad (1)$$

where θ_{y_i} represents the angle between the feature vector and the target weight vector, s denotes the feature scale, and m is a penalty parameter [7]. For our implementation we set m to 0.2 and s to 64.

3. ENSEMBLE SCORING AND FUSION

Anomaly detection in a "first-shot" scenario is inherently prone to high variance. To mitigate this, we employ a multi-backend ensemble, acknowledging that no single estimator can capture the multi-faced nature of mechanical failure.

3.1. Heterogeneous Backend Estimators

We utilize four distinct mathematical approaches to anomaly scoring, each configured with specific hyperparameters to capture different aspects of the latent embedding manifold:

- **k-nearest neighbors (KNN):** With $k = 6$. The Anomaly score is then calculated as a max-pooling of the neighbors [8].
- **Gaussian mixture models (GMMs):** Modeled using a mixture of $M = 6$ Gaussian components, this approach approximates the global density distribution of normal operation. Anomalies are identified by their low log-likelihood under this learned distribution [9].
- **AutoEncoders (AE):** Designed with a bottleneck dimension of 32 and a latent representation depth of 2 layers, this reconstruction-based backend identifies anomalies as samples with high residual reconstruction error, indicating that the input does not conform to the learned "normal" mapping. The AE are trained for 50 epochs with a learning rate $l = 1 \times 10^{-3}$ and a batch size of 64 [10].
- **global weighted ranking pooling (GWRP):** We employ a density-based estimator that computes a local outlier score using Local Density Neighbor (LDN) [11], which is subsequently refined through GWRP normalization. Utilizing a fixed weighting factor of $r = 0.98$ and a neighborhood radius defined by the local sample density, this approach smooths transient noise spikes while prioritizing consistent deviations from the manifold [12].

3.2. The Fusion Dynamics

The primary challenge in ensemble fusion is the disparity in score range and distribution. Raw scores from an AE (based on MSE) are inherently different from the log-likelihood scores of a GMM.

1. **Normalization Sensitivity:** Rank normalization is effective at handling the non-linear scale of these estimators, but it assumes that the distribution of anomalies and normal sounds is somewhat consistent across the batch. Our offline pipeline uses this to "smooth" the results [13].
2. **Online Calibration:** In the online scenario, we employ a sliding-window Z-score approach. By maintaining a running estimate of the mean and variance of the normal class, we ensure that the ensemble threshold is dynamic, allowing the system to adapt to subtle changes in machine behavior over time (e.g., thermal drift) without triggering false positives [14].

3.3. Min-Pooling Strategy

For the j-th model the pipeline generates four distinct anomaly scores denoted as $\{S_{NN}^{(j)}, S_{GWRP}^{(j)}, S_{GMM}^{(j)}, S_{AE}^{(j)}\}$. All scores are standardized using rank normalization or z-score normalization [14] [13]. The fused scores are then found by applying min-pooling on the average of the standardized scores as:

$$S_{fused} = \min_{i \in \{\text{NN,GWRP,GMM,AE}\}} \left(\frac{1}{N} \sum_{j=1}^{N} S_i^{(j)} \right) \tag{2}$$

By requiring a "unanimous" normal vote (where $\min$ effectively triggers an alert only when all models agree that the sample is atypical), we reduce the system's susceptibility to single-model variance. While this approach is conservative, it significantly increases the precision of the system in noisy environments.

4. PROPOSED SYSTEMS

To evaluate the impact of training data composition on system performance and generalizability, we developed four distinct pipeline configurations. All systems leverage a robust ensemble of pretrained transformer backbones specifically BEATs and SSLAM fine-tuned to extract highly discriminative features from the input audio.

The configurations differ based on two primary dimensions: the training data regime (Development set only vs. Development plus Additional set) and the inference constraint (Online vs. Offline). Yet the models trained on "DEV Only" includes 2x BEATs models (from different epochs) and 1x SSLAM model. The "DEV + ADD" only includes 1x BEATs and 1x SSLAM - due to time constraints. In the Offline pipelines, all four Scoring methods are uses. In Online GMM is not used as the Z-score could not handle the instability of the GMM.

4.1. System Configurations

We propose the following four pipeline submissions:

1. **Online Ensemble (DEV+ADD):** The counterpart to the offline pipeline, optimized for real-time inference using calibrated Z-score normalization.
2. **Offline Ensemble (DEV+ADD):** The primary pipeline, fine-tuned on both the development and the additional dataset, utilizing rank-based normalization to leverage global dataset statistics for maximum accuracy.
3. **Online Ensemble (DEV Only):** The real-time counterpart to the DEV-only configuration.
4. **Offline Ensemble (DEV Only):** A baseline configuration trained exclusively on the development set to isolate the performance gains introduced by the additional training data.

4.2. Rationale for Data Regimes

The inclusion of systems trained only on the development set allows us to quantify the "domain adaptation" capability of our ensemble. By comparing the *DEV+ADD* models against their *DEV-only* counterparts, we can assess whether the additional data provides valuable structural diversity or introduces noise that hampers the first-shot detection capability. This systematic comparison is essential for understanding how our fine-tuned ViT backbones handle the increased data variance presented in the 2026 challenge.

By evaluating these specific configurations, we aim to demonstrate that our fine-tuned transformer ensemble provides a robust, scalable framework that maintains high detection precision regardless of whether the model is exposed to the broader additional dataset or constrained to the development set.

5. RESULTS

The performance metrics for our proposed fusion pipelines are summarized in Table 1. For benchmark purposes, the results of the standard baseline models [15], Mahalanobis and mean square error (MSE), are provided in Table 2 to illustrate the relative performance gains achieved by our system.

Table 1: Performance Comparison of all Pipelines (AUC / pAUC). Best results in bold

	DEV Online		**DEV Offline**		**DEV+ADD Online**		**DEV+ADD Offline**	
Machine Type	**AUC**	**pAUC**	**AUC**	**pAUC**	**AUC**	**pAUC**	**AUC**	**pAUC**
ToyCar	78.28	56.00	**80.38**	**63.21**	78.02	53.84	77.55	57.18
ToyCarEmu	**81.32**	55.47	80.91	**71.13**	75.60	55.37	75.51	57.50
bearingEmu	62.86	**59.21**	**64.63**	57.84	64.38	57.37	64.62	53.82
fan	62.72	53.16	67.34	52.53	62.64	**55.95**	**72.25**	55.92
gearboxEmu	76.32	**68.53**	**79.66**	68.34	74.02	57.58	75.86	58.138
sliderEmu	**73.62**	**57.16**	70.91	55.45	68.44	54.74	66.33	53.58
valveEmu	**91.98**	**82.42**	89.61	81.45	91.08	77.68	89.94	77.00
Overall Mean	75.30	61.71	**76.21**	**64.28**	73.45	58.93	74.58	59.02
DCASE Score (Ω)	66.58		**68.60**		64.49		65.11	

Table 2: Performance Comparison of Baseline Models (AUC / pAUC).

	Mahalanobis		MSE	
Machine	**AUC**	**pAUC**	**AUC**	**pAUC**
ToyCar	0.6523	0.5825	0.5675	0.5403
ToyCarEmu	0.6806	0.5347	0.6541	0.5589
bearingEmu	0.6410	0.6042	0.6095	0.5985
fan	0.5255	0.5229	0.5420	0.5333
gearboxEmu	0.6361	0.5397	0.5901	0.5294
sliderEmu	0.5777	0.5036	0.5615	0.5038
valveEmu	0.5655	0.5020	0.6826	0.5508
Mean	0.6112	0.5414	0.6153	0.5450
Ω	0.5752		0.5791	

6. CONCLUSION

In conclusion, our pipelines successfully leverages supervised auxiliary classification to derive robust embeddings from pre-trained transformers. The combination of ArcFace optimization, diverse ensemble scoring, and adaptive normalization provides a high-precision framework for machine health monitoring that performs consistently across both online and offline industrial deployment scenarios.

7. REFERENCES

[1] IBM, "What is industry 4.0?" Available at https://www.ibm.com/think/topics/industry-4-0 (2026/02/17).

[2] D. Luo, Y. Quan, W. Xue, and F. Lin, "Motor fault detection based on sound signal," in *Proc. ICTCE*, 2024.

[3] T. Nishida, N. Harada, D. Takeuchi, D. Niizumi, K. Imoto, K. Dohi, H. Purohit, T. Endo, and Y. Kawaguchi, "Description and discussion on DCASE 2026 challenge task 2: Noise-aware unsupervised anomalous sound detection for machine condition monitoring," *arXiv:2606.01578*, 2026.

[4] S. Chen, Y. Wu, C. Wang, S. Liu, D. Tompkins, Z. Chen, and F. Wei, "BEATs: Audio pre-training with acoustic tokenizers," in *Proc. ICML*, 2022.

[5] T. Alex, S. Atito, A. Mustafa, M. Awais, and P. J. B. Jackson, "SSLAM: Enhancing self-supervised models with audio mixtures for polyphonic soundscapes," in *Proc. ICLR*, 2025.

[6] A. Chaudhuri, A. Dutta, T. Bui, and S. Georgescu, "A closer look at multimodal representation collapse," in *Proc. ICML*, 2025.

[7] J. Deng, J. Guo, J. Yang, N. Xue, I. Kotsia, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, 2022.

[8] Z. Zhang, "Introduction to machine learning: k-nearest neighbors," *Ann. Transl. Med.*, vol. 4, no. 11, 2016.

[9] D. Yu and L. Deng, *Gaussian Mixture Models*. London: Springer London, 2015, pp. 13–21. [Online]. Available: https://doi.org/10.1007/978-1-4471-5779-3_2

[10] D. Bank, N. Koenigstein, and R. Giryes, *Autoencoders*. Cham: Springer International Publishing, 2023, pp. 353–374. [Online]. Available: https://doi.org/10.1007/978-3-031-24628-9_16

[11] K. Wilkinghoff, H. Yang, J. Ebbers, F. G. Germain, G. Wichern, and J. Le Roux, "Local density-based anomaly score normalization for domain generalization," *IEEE Trans. Audio Speech Lang. Process.*, 2025.

[12] A. Kolesnikov and C. H. Lampert, "Seed, expand and constrain: Three principles for weakly-supervised image segmentation," in *Proc. ECCV*, 2016.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[14] N. Fei, Y. Gao, Z. Lu, and T. Xiang, "Z-score normalization, hubness, and few-shot learning," in *Proc. ICCV*, 2021.

[15] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, and M. Yasuda, "First-shot anomaly detection for machine condition monitoring: A domain generalization baseline," *Proc. EUSIPCO*, 2023.