GENREP WITH MULTI-BRANCH DUAL-CHANNEL INPUTS AND ADAPTIVE POOLING FOR TRAINING-FREE ANOMALOUS SOUND DETECTION

Technical Report

Chaoyong Huang[1], *Xiangyu Jing*[1], *Yuandong Luo*[1], *Hongqing Liu*[1]

[1]School of Communications and Information Engineering
Chongqing University of Posts and Telecommunications, Chongqing, China
hongqingliu@cqupt.edu.cn

ABSTRACT

This technical report describes our submission to the DCASE 2026 Challenge Task 2 on first-shot unsupervised anomalous sound detection (ASD) under domain shift. Our system builds on the GenRep framework proposed by Saengthong and Shinozaki, which performs training-free ASD using frozen embeddings from large-scale pre-trained audio encoders with k-nearest neighbor (kNN) scoring and domain-wise score normalization. We extend GenRep in three directions motivated by the dual-channel (near/far) audio setup newly introduced in DCASE 2026. First, we construct four input methods, including single-channel, far-weighted concatenation, inter-channel absolute difference, and STFT-domain soft-mask enhancement, each with parameter variants yielding seven configurations. Second, we incorporate embedding preprocessing and replace default mean pooling with adaptive temporal pooling strategies including generalized mean (GeM) pooling, relative deviation pooling (RDP), and hybrid RDP+GeM, following Wilkinghoff et al. Third, we perform per-encoder search over backend configurations and fuse complementary high-performing candidates via Z-score-aligned weighted score averaging. Our systems substantially outperform the official baselines on the development set.

***Index Terms*—** anomalous sound detection, training-free, GenRep, temporal pooling, multi-branch, dual-channel, domain generalization

1. INTRODUCTION

Anomalous sound detection (ASD) for machine condition monitoring identifies whether sounds emitted from a target machine are normal or anomalous using only normal reference data. The DCASE 2026 Challenge Task 2 [1, 2] continues the first-shot unsupervised ASD problem under domain generalization: systems must detect anomalies in entirely unseen machine types while remaining robust to domain shifts.

The task is built upon machine operating sound datasets designed for domain-shift and domain-generalization ASD, including ToyADMOS2 [3] and MIMII DG [4].

A key change in DCASE 2026 is the introduction of dual-channel audio. Each recording consists of a *near* channel (closer to the machine) and a *far* channel (captured from a distant microphone position). Therefore, effectively utilizing both channels is the central design challenge.

Recent approaches to domain-generalized ASD fall broadly into two categories. The first fine-tunes large-scale pre-trained audio encoders such as BEATs [5], EAT [6], and CED [7] with attribute classification, contrastive learning, or ArcFace objectives [8, 9, 10]. The second adopts a training-free paradigm, originating with GenRep [11]: Saengthong and Shinozaki proposed frozen BEATs embeddings with kNN search, MemMixup, and test-time domain Z-score normalization, achieving an score of 73.79% on DCASE 2023 Task 2 without any fine-tuning.

Saengthong and Shinozaki subsequently improved GenRep in their DCASE 2025 extension [12]: (1) replaced test-time Z-score with training-set domain-wise Z-score; (2) introduced Local Density Normalization (LDN) [13] and Domain-wise LDN (DLDN) to handle the source ($\sim$990) vs. target ($\sim$10) sample imbalance; (3) extended encoder coverage from BEATs to five models; (4) employed multi-encoder score ensembling.

Subsequently, Wilkinghoff et al. [14] systematically studied temporal pooling in training-free ASD, demonstrating that default mean pooling is suboptimal and proposing RDP and hybrid RDP+GeM, with consistent gains across five benchmark datasets. Their study also identified embedding preprocessing as crucial for high-dynamic-range encoders such as EAT.

Our system builds directly on the DCASE 2025 GenRep extension [12] and Wilkinghoff et al.'s pooling study [14]. We adopt the frozen-encoder kNN scoring framework with domain-wise Z-score / LDN / DLDN normalization from [12] and the preprocessing/pooling strategies from [14]. Our extensions specific to DCASE 2026 are:

- Four input methods (single-channel, far-weighted concatenation, inter-channel absolute difference, STFT-domain soft-mask enhancement) with parameter variants yielding seven configurations.
- Per-encoder search over preprocessing parameters and backend configurations; for pooling, we adopt optimal configurations from [14] after validation.
- Complementary score-level fusion via Z-score-aligned weighted averaging.

2. METHOD

2.1. System Overview

Figure 1 shows the overall pipeline of our dual-channel ASD system. The system builds on the DCASE 2025 GenRep extension [12], which combines frozen foundation encoders, domain-specific memory banks, kNN scoring, and domain-wise score normaliza-

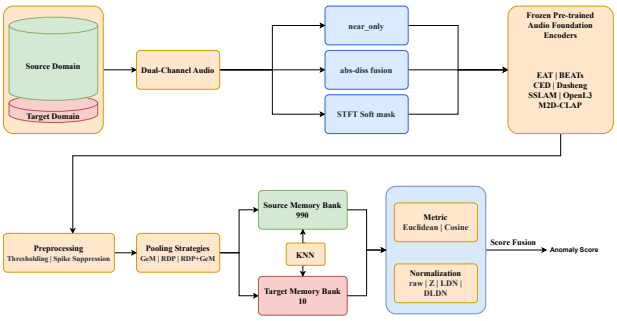

Figure 1: Overview of the proposed DCASE 2026 dual-channel ASD system.

tion. We further incorporate the preprocessing and pooling strategies proposed by Wilkinghoff et al. [14].

2.2. Base Framework

The DCASE 2025 GenRep extension [12] built upon the original GenRep [11] with three key improvements we directly adopt: (a) training-set domain-wise Z-score replacing test-time statistics; (b) LDN [13] and DLDN to handle source–target imbalance; (c) multi-encoder score ensembling. MemMixup from the original GenRep was dropped.

For a query embedding $\mathbf{z}$ and a normal memory bank $\mathcal{R}$, the single-bank kNN score is

$$d_k(\mathbf{z};\mathcal{R}) = \frac{1}{k}\sum_{\mathbf{r}\in\mathcal{N}_k(\mathbf{z},\mathcal{R})}\delta(\mathbf{z},\mathbf{r}), \tag{1}$$

where δ is Euclidean or cosine distance. For single-bank systems, $\mathcal{R}$ is the union of source and target normal embeddings. Domain-aware backends instead construct $\mathcal{R}_s$ and $\mathcal{R}_t$ separately and obtain $d_s = d_k(\mathbf{z};\mathcal{R}_s)$ and $d_t = d_k(\mathbf{z};\mathcal{R}_t)$. Domain-wise Z-score and DLDN calibrate the two banks independently before minimum or weighted combination. LDN calibrates a query distance using the local density of its neighboring normal embeddings.

2.3. Four Input Methods for Dual-Channel Audio

Embeddings are extracted independently from the near and far channels using the same frozen encoder, followed by frame preprocessing, temporal pooling, and L2 normalization, resulting in $\mathbf{z}_{\text{near}}$ and $\mathbf{z}_{\text{far}}$.

Method 1 – Single-Channel (near_only).
Only the near-channel embedding is used:

$$\mathbf{z} = \mathbf{z}_{\text{near}}. \tag{2}$$

This serves as the cleanest single-channel baseline.

Method 2 – Far-Weighted Concatenation (n_0.5f, n_f).
The far-channel embedding is scaled before concatenation:

$$\mathbf{z} = \mathrm{L2Norm}\Big(\mathrm{concat}(\mathbf{z}_{\text{near}}, \beta\mathbf{z}_{\text{far}})\Big), \tag{3}$$

where $\beta \in \{0.5,\ 1.0\}$ controls the contribution of the far channel.

Method 3 – Inter-Channel Absolute Difference (absdif_1, absdif_0.5).

An explicit channel-difference representation is first computed as

$$\mathbf{z}_{\text{diff}} = \mathrm{L2Norm}\left(|\mathbf{z}_{\text{near}} - \mathbf{z}_{\text{far}}|\right). \tag{4}$$

The final representation is

$$\mathbf{z} = \mathrm{L2Norm}\Big(\mathrm{concat}(\mathbf{z}_{\text{near}}, \mathbf{z}_{\text{far}}, \gamma\mathbf{z}_{\text{diff}})\Big), \tag{5}$$

where $\gamma \in \{1.0,\ 0.5\}$ controls the strength of the difference branch.

Method 4 – STFT Soft-Mask Enhancement (sfsoft_a03f05, sfsoft_a05f07).

Instead of manipulating embeddings directly, this method enhances the near-channel waveform in the time-frequency domain before feature extraction.

Let $N(f,t)$ and $F(f,t)$ denote the STFTs of the near and far channels, respectively.

A frequency-wise transfer estimate is first computed as

$$H(f) = \frac{\sum_t N(f,t)\overline{F(f,t)}}{\sum_t |F(f,t)|^2 + \varepsilon}. \tag{6}$$

The estimated far-channel leakage is

$$P(f,t) = H(f)F(f,t), \tag{7}$$

and its relative energy ratio is

$$R(f,t) = \frac{|P(f,t)|^2}{|N(f,t)|^2 + \varepsilon}. \tag{8}$$

A soft suppression mask is then computed as

$$M(f,t) = \mathrm{clip}\left(1 - \alpha\frac{R(f,t)}{1+R(f,t)}, \mathrm{floor}, 1.0\right). \tag{9}$$

The enhanced spectrum is obtained by

$$E(f,t) = M(f,t)N(f,t), \tag{10}$$

followed by waveform reconstruction:

$$\mathbf{x}_{\text{enh}} = \mathrm{ISTFT}(E). \tag{11}$$

The enhanced waveform is re-encoded using the same frozen encoder:

$$\mathbf{z}_{\text{enh}} = \phi(\mathbf{x}_{\text{enh}}). \tag{12}$$

For anomaly scoring, score-level fusion is employed after each component score is aligned using training-normal score statistics:

$$s_{\text{final}} = 0.6\,\widetilde{s}_{\text{near}} + 0.2\,\widetilde{s}_{\text{enh}} + 0.2\,\widetilde{s}_{\text{stereo}}. \tag{13}$$

Two parameter configurations are evaluated: ($\alpha = 0.3$, $\mathrm{floor} = 0.5$) for mild suppression and ($\alpha = 0.5$, $\mathrm{floor} = 0.7$) for stronger suppression.

Unlike embedding subtraction methods, the proposed approach operates entirely in the STFT domain. The enhanced waveform is reconstructed and passed through the full encoder pipeline, allowing the pretrained encoder to extract features from the modified acoustic signal.

2.4. Frozen Audio Encoders

We employ eight frozen pre-trained audio encoders (Table 1). All parameters are frozen.

Table 1: Frozen pre-trained audio encoders.

Encoder	Arch.	Pre-train	Params
EAT-large	EAT	AS-2M + SSL	88M
EAT-AS2M	EAT	AS-2M	88M
BEATs	Trans.	AudioSet	90M
CED-base	ViT	AudioSet	85M
M2D-CLAP	CLAP	AS+LAION	300M
SSLAM	Mamba	AudioSet	100M
OpenL3	CNN	SSL	5M
Dasheng	Trans.	Large-scale	90M

2.5. Embedding Preprocessing

As reported by Wilkinghoff et al. [14], encoders with high-dynamic-range frame activations can benefit from low-value hard thresholding and spike suppression. For frame component $x_{t,d}$, values below τ are set to zero, while values above ρ are replaced by $\tanh(x_{t,d})$. We search six configurations: $(\text{None}, \text{None})$, $(\text{None}, 0.2)$, $(\text{None}, 0.3)$, $(\text{None}, 0.4)$, $(0.10, 0.3)$, and $(0.15, 0.3)$. The effect is particularly strong for EAT.

2.6. Temporal Pooling

The original GenRep framework employs mean pooling for temporal aggregation. However, Wilkinghoff et al. [14] demonstrated that pooling selection has a significant impact on anomaly detection performance and proposed encoder-specific optimal pooling strategies. Following their findings, we evaluate several pooling configurations on representative encoders (EAT, BEATs, and Dasheng) and subsequently adopt the recommended pooling strategy for each encoder.

Generalized Mean Pooling (GeM) [15]:

$$\mathrm{GeM}(\mathbf{x}) = \left(\frac{1}{T} \sum_{t=1}^{T} \max(0, \mathbf{x}_t)^p \right)^{1/p}, \tag{14}$$

where $p \in \{1, 3, 9\}$ controls the pooling sharpness.

Relative Deviation Pooling (RDP) [14]:

$$\begin{aligned} d_t &= \|\mathbf{x}_t - \overline{\mathbf{x}}\|_2, \qquad \hat{d}_t = \frac{d_t}{\max_u d_u + \varepsilon}, \\ w_t &= \frac{(1+\hat{d}_t)^\alpha}{\sum_u (1+\hat{d}_u)^\alpha}, \qquad \mathrm{RDP}(\mathbf{x}) = \sum_t w_t \mathbf{x}_t. \end{aligned} \tag{15}$$

where $\hat{d}_t$ is the normalized relative deviation from the temporal mean, and $\alpha \in \{1, 9, 16, 19, 20\}$ controls the emphasis on deviating frames.

RDP+GeM [14]:

The hybrid applies the normalized RDP weights inside GeM aggregation:

$$\mathrm{RDPGeM}(\mathbf{x}) = \left(\sum_t w_t \max(0, \mathbf{x}_t)^p \right)^{1/p}. \tag{16}$$

2.7. Backend Scoring

We adopt the domain-generalized kNN framework from [12] and search over backend configurations. LDN and DLDN are directly adopted from [12, 13].

Table 2: Backend search space used for system selection.

Parameter	**Search Space**
Distance	{Euclidean, Cosine}
Single-bank k	$\{1, 3, 5, 10, 20\}$
Calibration	{Raw, LDN}
Domain-aware	{domain-wise Z, DLDN, weighted DB-LDN}

2.8. Score-Level Fusion

Complementary systems are combined using weighted score fusion. For system i, training-normal scores provide mean μ_i and standard deviation σ_i, and

$$\widetilde{s}_i = \frac{s_i - \mu_i}{\sigma_i + \varepsilon}. \tag{17}$$

The fused score is

$$s_{\text{fused}} = \sum_i \lambda_i \widetilde{s}_i, \quad \lambda_i \geq 0, \quad \sum_i \lambda_i = 1. \tag{18}$$

This cross-system alignment is separate from any normalization performed inside an individual kNN backend.

3. EXPERIMENTAL SETUP

The DCASE 2026 Task 2 development set [2] contains seven machine types derived from the ToyADMOS2 [3] and MIMII DG [4] dataset families: *ToyCar*, *ToyCarEmu*, *bearingEmu*, *fan*, *gearboxEmu*, *sliderEmu*, and *valveEmu*. Each machine type provides approximately 990 source-domain normal samples and 10 target-domain normal samples in a dual-channel (near/far) recording setup.

For each machine type, the machine score is the harmonic mean of source-domain AUC, target-domain AUC, and pAUC:

$$\Omega_m = \frac{3}{\frac{1}{\mathrm{AUC}_{s,m}} + \frac{1}{\mathrm{AUC}_{t,m}} + \frac{1}{\mathrm{pAUC}_m}}. \tag{19}$$

The overall score reported by the challenge is the harmonic mean over the three metrics of all development machine types.

All foundation encoders are used in a frozen inference-only mode without parameter updates. For the STFT soft-mask enhancement, a 1024-point Hann window and a hop size of 256 samples are employed. All clip-level embeddings are L2-normalized prior to anomaly scoring, and kNN distances are computed using exact nearest-neighbor search.

4. RESULTS

4.1. Best Single-System Results

Table 3 summarizes the best single-system result per encoder. The best fusion combines EAT-large, Dasheng, SSLAM, EAT-AS2M, and OpenL3 with weights 0.55, 0.20, 0.05, 0.05, and 0.15, respectively. Before fusion, each subsystem score is aligned using its training-normal score statistics as described in Section 2. All results in this section are measured on the development set.

Table 3: Best single-system results per encoder.

Encoder	Best Input	Best Pooling	Best Backend	Score
EAT-large	near_only	RDP+GeM(α=1,p=3)	Cos k=1 raw	**60.93**
Dasheng	sfsoft_a05f07	RDP(α=20)	Euc k=3 LDN	60.06
OpenL3	sfsoft_a03f05	GeM(p=9)	Cos k=3 LDN	59.48
EAT-AS2M	near_only	RDP+GeM(α=1,p=3)	Euc k=1 raw	59.33
SSLAM	absdif_0.5	RDP+GeM(α=9,p=3)	Cos k=5 LDN	59.06
CED-base	sfsoft_a05f07	RDP+GeM(α=9,p=3)	Cos k=20 LDN	58.46
M2D-CLAP	sfsoft_a05f07	RDP+GeM(α=9,p=3)	Cos k=20 LDN	58.28
BEATs	n_0.5f	RDP(α=19)	Cos k=3 LDN	57.65

Table 4: Comparison with official baselines.

Method	AUC_s	AUC_t	pAUC	Score
AE-MSE (official) [16]	67.46	52.74	54.50	56.41
AE-MAHALA (official) [16]	67.16	55.08	54.14	57.33
Our EAT-large (best single)	68.95	64.19	54.72	60.93
Our best fusion	**71.53**	**65.74**	**59.16**	**63.41**

Table 5: Per-machine comparison with official baselines on the development set (%).

Machine	System	AUC_s	AUC_t	pAUC
ToyCarEmu	MAHALA	69.49	66.62	53.47
	MSE	**69.62**	61.20	**55.89**
	Ours	66.70	**69.82**	50.11
ToyCar	MAHALA	77.28	53.17	58.25
	MSE	75.62	37.87	54.03
	Ours	**80.98**	**79.72**	**62.11**
bearingEmu	MAHALA	**65.92**	62.28	60.42
	MSE	62.34	59.56	59.85
	Ours	63.18	**63.88**	**61.26**
fan	MAHALA	60.00	45.09	52.29
	MSE	**61.45**	46.94	**53.33**
	Ours	59.70	**49.88**	51.37
gearboxEmu	MAHALA	**74.48**	**52.74**	**53.97**
	MSE	68.23	49.78	52.94
	Ours	72.12	48.30	52.89
sliderEmu	MAHALA	66.36	49.18	50.36
	MSE	67.25	45.05	50.38
	Ours	**73.54**	**61.32**	**56.84**
valveEmu	MAHALA	56.60	56.50	50.20
	MSE	67.74	68.78	55.08
	Ours	**84.48**	**87.26**	**79.53**

4.2. Per-Machine Comparison with Official Baselines

The fusion yields particularly strong improvements for ToyCar, sliderEmu, and valveEmu. It also improves target-domain AUC for ToyCarEmu, bearingEmu, and fan. However, fan and gearboxEmu remain difficult, while ToyCarEmu pAUC does not exceed the official baselines.

4.3. Input Method Analysis

The optimal input method is **encoder-dependent**. The EAT family prefers near_only, BEATs selects far-weighted concatenation, and SSLAM selects absdif_0.5. Dasheng, CED-base, and M2D-CLAP select sfsoft_a05f07, whereas OpenL3 selects sfsoft_a03f05. These results show that the far channel is useful for several encoders, but no single dual-channel construction dominates all models.

4.4. Pooling and Preprocessing

The EAT family, SSLAM, CED-base, and M2D-CLAP select RDP+GeM; BEATs and Dasheng select RDP; and OpenL3 selects GeM. EAT-large and EAT-AS2M use low-value thresholding at 0.15 and spike suppression at 0.3, while the selected Dasheng and OpenL3 systems do not use activation preprocessing. This supports encoder-specific rather than globally fixed pooling and preprocessing.

4.5. Backend Observations

Cosine distance is selected by six of the eight best single systems. LDN is selected by six systems, while both EAT variants prefer raw $k = 1$ scoring. The optimal k ranges from 1 to 20, confirming that the backend should be selected per encoder.

5. CONCLUSION

We presented a training-free ASD system for DCASE 2026 Task 2, built on the DCASE 2025 GenRep extension [12] and Wilkinghoff et al.'s pooling study [14]. Our extensions address the dual-channel setting with four input methods (seven configurations), per-encoder search, and complementary score fusion. All eight selected single systems exceed the strongest official baseline overall score on the development set. EAT-large achieves the best single-system score of 60.93%, and the complementary five-system fusion further improves the score to 63.41%.

6. REFERENCES

[1] T. Nishida, N. Harada, D. Niizumi, D. Albertini, R. Sannino, S. Pradolini, F. Augusti, K. Imoto, K. Dohi, H. Purohit, T. Endo, and Y. Kawaguchi, "Description and discussion on DCASE 2025 challenge task 2: First-shot unsupervised anomalous sound detection for machine condition monitoring," DCASE2025 Challenge, Tech. Rep., 2025, arXiv:2506.10097.

[2] T. Nishida, N. Harada, D. Niizumi, *et al.*, "Description and discussion on DCASE 2026 challenge task 2," DCASE2026 Challenge, Tech. Rep., 2026.

[3] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, M. Yasuda, and S. Saito, "ToyADMOS2: Another dataset of miniature-machine operating sounds for anomalous sound detection under domain shift conditions," in *Proc. DCASE Workshop*, 2021, pp. 1–5.

[4] K. Dohi, T. Nishida, H. Purohit, R. Tanabe, T. Endo, M. Yamamoto, Y. Nikaido, and Y. Kawaguchi, "MIMII DG: Sound dataset for malfunctioning industrial machine investigation and inspection for domain generalization task," in *Proc. DCASE Workshop*, 2022.

[5] S. Chen, Y. Wu, C. Wang, S. Liu, D. Tompkins, Z. Chen, W. Che, X. Yu, and F. Wei, "BEATs: Audio pre-training with acoustic tokenizers," in *Proc. ICML*, vol. 202, 2023, pp. 5178–5193.

[6] W. Chen, Y. Liang, Z. Ma, Z. Zheng, and X. Chen, "EAT: Self-supervised pre-training with efficient audio transformer," in *Proc. IJCAI*, 2024.

[7] H. Dinkel, Y. Wang, Z. Yan, J. Zhang, and Y. Wang, "CED: Consistent ensemble distillation for audio tagging," 2023.

[8] L. Wang, "Pre-trained model enhanced anomalous sound detection system for DCASE2025 task2," DCASE2025 Challenge, Tech. Rep., 2025.

[9] K. Ozeki, T. Shiraga, T. Masuzaki, N. Tanaka, and T. Kuriyama, "Anomalous sound detection method using contrastive learning," DCASE2025 Challenge, Tech. Rep., 2025.

[10] S. Huang and L. He, "XJU system for first-shot unsupervised anomalous sound detection," DCASE2025 Challenge, Tech. Rep., 2025.

[11] P. Saengthong and T. Shinozaki, "Deep generic representations for domain-generalized anomalous sound detection," in *Proc. ICASSP*, 2025, arXiv:2409.05035, 2024.

[12] ——, "GenRep for first-shot unsupervised anomalous sound detection of DCASE 2025 challenge," DCASE2025 Challenge, Tech. Rep., 2025.

[13] K. Wilkinghoff, H. Yang, J. Ebbers, F. G. Germain, G. Wichern, and J. L. Roux, "Keeping the balance: Anomaly score calculation for domain generalization," in *Proc. ICASSP*, 2025.

[14] K. Wilkinghoff, S. Yadav, and Z.-H. Tan, "Temporal pooling strategies for training-free anomalous sound detection with self-supervised audio embeddings," 2026.

[15] F. Radenović, G. Tolias, and O. Chum, "Fine-tuning CNN image retrieval with no human annotation," *IEEE Trans. PAMI*, vol. 41, no. 7, pp. 1655–1668, 2018.

[16] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, and M. Yasuda, "First-shot anomaly detection for machine condition monitoring: A domain generalization baseline," in *Proc. EUSIPCO*, 2023, pp. 191–195.