MULTI-CHANNEL BRANCH CONSTRUCTION AND GRAPH-REFINED MEMORY BANKS FOR DCASE 2026 CHALLENGE TASK 2

Technical Report

Zhang Cheng, Masashi Unoki

Graduate School of Advanced Science and Technology,
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikawa 923–1292 Japan
{s2510417, unoki}@jaist.ac.jp

ABSTRACT

This technical report presents a method for DCASE 2026 Challenge Task 2, which focuses on unsupervised anomalous sound detection under noisy two-channel recording conditions. The key challenge is how to effectively exploit multi-channel information and improve memory-bank based anomaly scoring. We exploit the multi-channel nature of the dataset by constructing multiple channel embedding branches, including single-channel, mono, and inter-channel difference representations. To improve nearest-neighbor based anomaly scoring, we refine source and target memory banks with graph smoothing, align test samples to the refined memory structure at inference time, and apply local-density normalized KNN scoring. Finally, scores from multiple pretrained encoder branches are fused at the score level. On the DCASE 2026 development set, the proposed system achieves an official score of 0.661203, outperforming both the official baseline and the GenRepASD baseline. Experimental results show that multi-channel branch construction is the main contributor, while graph refinement, test-time graph alignment, and density-normalized KNN score provide additional consistent gains. Our source code is available at `https://github.com/infolence/DCASE2026_Task2.git`.

***Index Terms*—** Anomalous sound detection, Machine condition monitoring, DCASE challenge

1. INTRODUCTION

Anomalous sound detection (ASD) aims to identify abnormal machine conditions from acoustic signals [1][2]. It is an important technique for machine condition monitoring, especially in scenarios where collecting sufficient anomalous samples is difficult or impractical [3]. DCASE Task 2 focuses on unsupervised ASD, where only normal training samples are available and the system must assign anomaly scores to unseen test recordings [4]. This setting is challenging because the model needs to capture the distribution of normal operating sounds while remaining sensitive to subtle acoustic deviations caused by abnormal machine states.

Recent approaches have shown that pretrained audio encoders can provide strong general-purpose representations for ASD [5]. Instead of training a task-specific model from scratch, GenRepASD [6] methods use frozen audio encoders to extract embeddings and perform memory-bank based nearest-neighbor scoring. These methods are simple and effective, but they mainly rely on the quality of frozen representations and the raw distance between a test embedding and normal training embeddings. As a result, the scoring process may be sensitive to domain mismatch, sparse target-domain samples, and local variations in the normal embedding distribution.

The DCASE 2026 Task 2 dataset introduces multi-channel recordings, which provide additional spatial and inter-channel information compared with single-channel audio [7][8]. However, directly averaging the two channels into a mono waveform may discard useful discrepancy cues between channels. Motivated by this observation, we construct channel branches from the two-channel input. In addition to the first-channel and mono embeddings, we use the absolute difference between channel embeddings to capture inter-channel acoustic discrepancies. These complementary branches are later combined through score-level fusion.

Beyond channel branch construction, we also improve the memory-bank scoring process. We refine source and target normal memory banks using graph smoothing, so that each normal embedding can incorporate information from its local neighbors. At inference time, each test sample is softly attached to the refined memory structure before scoring. Finally, we introduce local-density normalized KNN scoring, which adjusts the test-to-memory distance according to the density around the nearest normal memory node. This allows the system to distinguish deviations in dense normal regions from those in sparse regions more effectively.

Our contributions are summarized as follows:

- We extend a frozen-encoder memory-bank ASD framework with channel branch construction for multi-channel machine sounds.
- We improve memory-bank based anomaly scoring with graph-based memory refinement, Test-time Graph Alignment, and local-density normalized KNN scoring.

2. DATASET

We use only the official dataset provided for DCASE 2026 Task 2. The task dataset consists of a development dataset, an additional training dataset, and an evaluation dataset. The development dataset contains two-channel recordings from seven machine types, with each clip lasting 10 or 12 seconds and including both target-machine sounds and environmental noise. Channel 1 is recorded by a microphone placed near the target machine, whereas channel 2 is recorded farther away. The training data are organized into source

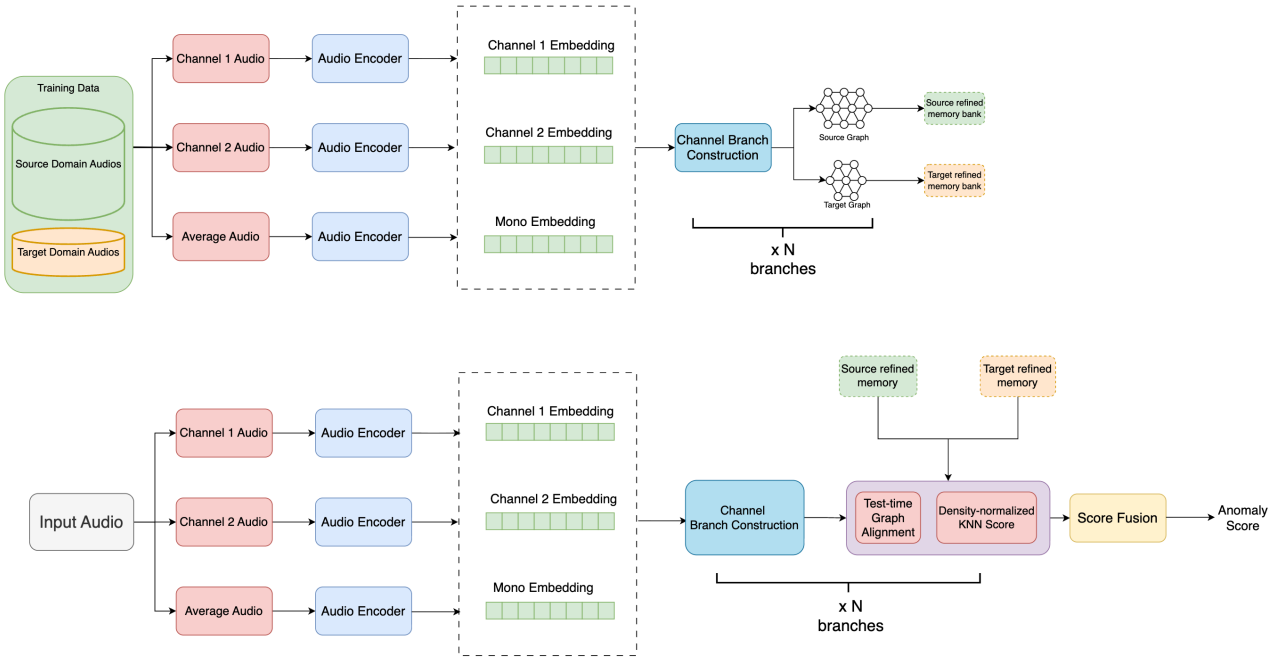

Figure 1: Overview of the proposed model.

and target domains, and only normal samples are provided for training, while the test data contain both normal and anomalous samples. The additional training dataset and the evaluation dataset also contain machine operating sounds, but their machine types are different from those in the development dataset. In this work, we use the development set to evaluate the proposed system and compare it with the official baseline [9].

3. PROPOSED METHOD

3.1. Overview

Our system follows a frozen-encoder memory-bank based anomaly detection framework, as show in Fig. 1. Given input audio clip, we first extract embedding using pretrained audio encoders without updating their parameters. The normal training samples are then stored as source and target memory banks. To get the information of nearest-neighbor embedding, we further refine the memory banks with graph smoothing, attach each test sample to the refined graph at inference time, and finally apply local-density normalized KNN scoring.

3.2. Channel Branch Construction

Since the DCASE 2026 data contain two-channel recordings, we explicitly construct multi-channel branches. Let x_1 and x_2 denote the two input channels. For a frozen encoder $f(\cdot)$, the corresponding embeddings are

$$z_1 = f(x_1), \qquad z_2 = f(x_2), \tag{1}$$

where x_1 and x_2 are the first and second audio channels, $f(\cdot)$ is a frozen pretrained audio encoder, and z_1 and z_2 are the extracted embeddings.

To exploit inter-channel discrepancy, we construct an absolute-difference branch:

$$z_{\text{diff}} = |z_1 - z_2|, \tag{2}$$

where z_{diff} denotes the inter-channel difference embedding, and $|\cdot|$ is the element-wise absolute value.

Each branch independently produces an anomaly score, and the final anomaly score is obtained by weighted score-level fusion:

$$s_{\text{BEATs}} = w_1 s_{\text{BEATs\_ch1}} + w_2 s_{\text{BEATs\_diff}}, \tag{3}$$

$$s_{\text{CED}} = w_3 s_{\text{CED\_diff}} + w_4 s_{\text{CED\_mono}}, \tag{4}$$

$$s_{\text{final}} = s_{\text{BEATs}} + s_{\text{CED}}, \tag{5}$$

where s_{final} is the final anomaly score, $s_{\text{BEATs\_ch1}}$, $s_{\text{BEATs\_diff}}$, $s_{\text{CED\_diff}}$, and $s_{\text{CED\_mono}}$ are branch-level anomaly scores, and w_1, w_2, w_3, w_4 are their fusion weights.

3.3. Graph Refinement for the Memory Bank

For each branch, the source-domain and target-domain normal samples are separated into two memory banks:

$$H_s = \{h_i^s\}_{i=1}^{N_s}, \qquad H_t = \{h_i^t\}_{i=1}^{N_t}, \tag{6}$$

where H_s and H_t are the source and target memory banks, h_i^s and h_i^t are normal training embeddings, and N_s and N_t are the numbers of source and target normal samples.

After adding self-loops, we apply GCN symmetric normalization [10]:

$$\hat{A} = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}, \tag{7}$$

where A is the adjacency matrix of the KNN graph [11], D is the degree matrix, and $\hat{A}$ is the normalized adjacency matrix.

Table 1: Overall comparison on the DCASE 2026 development set.

Dataset	Method	$\mathrm{AUC}_{\mathrm{source}}$ Mean	$\mathrm{AUC}_{\mathrm{target}}$ Mean	pAUC Mean	Official Score
ToyCarEmu	Official baseline	**69.49%**	66.62%	53.47%	62.37%
	GenRepASD	65.57%	**79.28%**	53.58%	**64.48%**
	Proposed model	61.76%	76.38%	**53.63%**	62.59%
ToyCar	Official baseline	77.28%	53.17%	58.25%	61.33%
	GenRepASD	**79.84%**	74.18%	**60.58%**	70.57%
	Proposed model	77.18%	**80.82%**	58.79%	**70.86%**
bearingEmu	Official baseline	65.92%	62.28%	60.42%	62.79%
	Backbone	61.86%	57.54%	59.47%	59.57%
	Proposed model	**66.82%**	**62.30%**	**60.42%**	**63.07%**
fan	Official baseline	61.45%	46.94%	53.33%	53.26%
	GenRepASD	67.70%	51.33%	54.42%	57.00%
	Proposed model	**69.24%**	**59.98%**	**54.74%**	**60.75%**
gearboxEmu	Official baseline	**74.48%**	52.74%	53.97%	58.92%
	GenRepASD	68.82%	52.22%	51.16%	56.36%
	Proposed model	70.24%	**70.86%**	**62.58%**	**67.68%**
sliderEmu	Official baseline	66.36%	49.18%	50.36%	54.29%
	GenRepASD	61.06%	55.70%	52.58%	56.23%
	Proposed model	**74.62%**	**67.18%**	**58.58%**	**66.14%**
valveEmu	Official baseline	67.74%	68.78%	55.08%	63.22%
	GenRepASD	69.68%	**78.84%**	60.58%	68.90%
	Proposed model	**85.40%**	74.14%	**64.53%**	**73.72%**

Table 2: Average performance across machine types on the DCASE 2026 development set.

Method	$\mathrm{AUC}_{\mathrm{source}}$ Mean	$\mathrm{AUC}_{\mathrm{target}}$ Mean	pAUC Mean	Official Score
Official baseline	68.61%	55.99%	54.81%	59.20%
GenRepASD	67.10%	62.44%	55.85%	61.45%
Proposed model	**71.50%**	**69.53%**	**58.81%**	**66.12%**

The memory embeddings are refined by one-step graph propagation [12]:

$$H' = (1-\alpha)H + \alpha\hat{A}H, \tag{8}$$

where H is the original memory bank, H' is the refined memory bank, α is the graph smoothing coefficient, and $\hat{A}H$ aggregates neighbor information.

This operation is applied separately to source and target memory banks:

$$H'_s = (1-\alpha)H_s + \alpha\hat{A}_s H_s, \tag{9}$$

$$H'_t = (1-\alpha)H_t + \alpha\hat{A}_t H_t, \tag{10}$$

where H'_s and H'_t are refined source and target memory banks, and $\hat{A}_s$ and $\hat{A}_t$ are the normalized adjacency matrices for the source and target graphs.

3.4. Test-time Graph Alignment

The refined source and target memory banks are combined as:

$$H'_{\mathrm{all}} = H'_s \cup H'_t, \tag{11}$$

where H'_{all} denotes the union of the refined source memory bank H'_s and refined target memory bank H'_t.

For each test embedding, we perform one-step Test-time Graph Alignmentment:

$$h'_y = (1-\alpha_{\mathrm{test}})h_y + \alpha_{\mathrm{test}} m_{\mathrm{NN}}, \tag{12}$$

where h_y is the original test embedding, h'_y is the updated test embedding, m_{NN} is the nearest refined memory node, and α_{test} controls the attachment strength.

3.5. Density-normalized KNN Score

For each memory node, local density is computed as:

$$\rho_i = \frac{1}{k_\rho} \sum_{j \in \mathcal{N}(i)} \|m_i - m_j\|_2, \tag{13}$$

where ρ_i is the local density term of memory node m_i, $\mathcal{N}(i)$ is the set of its k_ρ nearest memory neighbors, and $\|\cdot\|_2$ denotes Euclidean distance.

Table 3: Ablation study of the proposed system on the DCASE 2026 development set.

Setting	AUC_s	AUC_t	pAUC	Official Score
Channel 1 only	68.52%	67.42%	55.91%	63.41%
Channel 2 only	58.28%	60.49%	54.45%	57.63%
mono Channel only	67.01%	68.15%	55.77%	63.12%
w/o Graph Refinement	70.35%	69.91%	57.90%	65.52%
w/o Test-time Graph Alignment	71.14%	68.75%	57.61%	65.27%
w/o Density Norm.	70.53%	66.37%	58.14%	64.59%
Proposed method	**71.50%**	**69.53%**	**58.81%**	**66.12%**

The density-normalized anomaly score is:

$$s(y) = \frac{\|h'_y - m_{\mathrm{NN}}\|_2}{(\rho_{\mathrm{NN}} + \epsilon)^{\gamma}}, \tag{14}$$

where $s(y)$ is the anomaly score of test sample y, h'_y is the attached test embedding, m_{NN} is its nearest memory node, ρ_{NN} is the local density of that nearest memory node, ϵ is a small constant for numerical stability, and γ controls the strength of density normalization.

Finally, the branch-level GenRep-style score is:

$$s_{\mathrm{branch}}(y) = \min(s_s(y), s_t(y)), \tag{15}$$

where $s_s(y)$ and $s_t(y)$ are the source-side and target-side anomaly scores, respectively, and $s_{\mathrm{branch}}(y)$ is the final score of one branch.

4. EXPERIMENT SETUP

For memory-bank scoring, we use top-1 nearest-neighbor distance. The source and target memory banks are refined separately using a KNN graph with $k = 5$ and graph smoothing coefficient $\alpha = 0.2$. During inference, each test sample is attached to its nearest refined memory node with $\alpha_{\mathrm{test}} = 0.3$.

For density-normalized scoring, we compute local density using $k_\rho = 3$ nearest memory neighbors and set $\gamma = 0.5$. The final system uses score-level fusion over four branches: BEATs channel-1, BEATs absolute-difference, CED-tiny absolute-difference, and CED-tiny mono. The corresponding fusion weights are set to 1.00, 1.75, 1.25, and 0.50. After branch-level scoring, domain-wise Z-score normalization is applied separately to source-domain and target-domain test samples before score fusion.

5. EVALUATION METRICS

System performance is evaluated using AUC and pAUC. For each machine type, AUC is computed separately on the source-domain and target-domain test samples, denoted as AUC_s and AUC_t, respectively. pAUC is computed over the range with a maximum false positive rate of 0.1. The final official score is defined as the harmonic mean of AUC_s, AUC_t, and pAUC:

$$\text{Official Score} = H_{\mathrm{mean}}\left(\mathrm{AUC}_s, \mathrm{AUC}_t, \mathrm{pAUC}\right), \tag{16}$$

where $H_{\mathrm{mean}}(\cdot)$ denotes the harmonic mean.

6. MAIN RESULTS

Table 1 presents the performance of different models across seven machine types on the DCASE 2026 development set. Compared with the official baseline, the proposed model improves the Official Score on six out of the seven datasets, demonstrating the effectiveness of the method across diverse machine conditions. Although the Official Score on ToyCarEmu is slightly lower than that of GenRepASD, the proposed model still achieves the best pAUC performance on this subset. As summarized by the average performance in Table 2, the proposed model achieves the best overall performance with an Official Score of 66.12%, outperforming the official baseline by 6.92%. Notably, the AUC_{target} Mean, which represents cross-domain generalization capability, yields a significant absolute improvement of 13.54%, further demonstrating its exceptional domain adaptability.

7. ABLATION STUDY

Table 3 presents the ablation study results of the proposed system on the DCASE 2026 development set. Compared with the full proposed method, relying solely on any single input channel (Channel 1, Channel 2, or a mono channel) leads to a performance drop, demonstrating the effectiveness of utilizing full multi-channel information, with Channel 2 alone showing particularly weak performance. Similarly, removing key architectural components (such as the graph structure and Test-time Graph Alignment) also limits the final accuracy, indicating their substantial contributions to structural feature learning and domain adaptation. Notably, among the various modules, the absence of density normalization results in the most significant performance degradation, proving its critical role in maintaining overall system performance. In summary, the ablation results fully validate the necessity of each individual component; it is through the effective integration of multi-channel information, graph-based structural learning, and density normalization that the complete proposed system achieves the highest accuracy.

8. CONCLUSION

In this work, we propose a method for DCASE 2026 Task 2. The proposed system using multi-channel recordings through channel branches, refines normal memory embeddings with graph smoothing, and uses test-time attachment with local-density normalized KNN scoring. Experiments on the development set show that our system achieves an official score of 66.12%, outperforming both the official baseline and the reproduced GenRepASD baseline. Ablation results show that multi-channel branch construction contributes the most, while graph-based refinement and Density-normalized KNN scoring provide further gains. Future work will explore adaptive graph construction, improved score fusion, and more robust calibration.

9. ACKNOWLEDGMENT

This work was supported by JSPS KAKENHI (25H01139).

10. REFERENCES

[1] 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.

[2] Y. Koizumi, Y. Kawaguchi, K. Imoto, T. Nakamura, Y. Nikaido, R. Tanabe, H. Purohit, K. Suefusa, T. Endo, M. Yasuda, *et al.*, "Description and discussion on dcase2020 challenge task2: Unsupervised anomalous sound detection for machine condition monitoring," *arXiv preprint arXiv:2006.05822*, 2020.

[3] T. Nishida, N. Harada, D. Niizumi, D. Albertini, R. Sannino, S. Pradolini, F. Augusti, K. Imoto, K. Dohi, H. Purohit, *et al.*, "Description and discussion on dcase 2025 challenge task 2: First-shot unsupervised anomalous sound detection for machine condition monitoring," *arXiv preprint arXiv:2506.10097*, 2025.

[4] 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," *In arXiv e-prints: 2606.01578*, 2026.

[5] K. Wilkinghoff, "Self-supervised learning for anomalous sound detection," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 276–280.

[6] P. Saengthong and T. Shinozaki, "Genrep for first-shot unsupervised anomalous sound detection of dcase 2025 challenge," *DCASE2025 Challenge, Barcelona, Spain, Tech. Rep*, 2025.

[7] 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 *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, Barcelona, Spain, November 2021, pp. 1–5.

[8] 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 *Proceedings of the 7th Detection and Classification of Acoustic Scenes and Events 2022 Workshop (DCASE2022)*, Nancy, France, November 2022.

[9] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, and M. Yasuda, "First-shot anomaly detection for machine condition monitoring: A domain generalization baseline," *Proceedings of 31st European Signal Processing Conference (EUSIPCO)*, pp. 191–195, 2023.

[10] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[11] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[12] F. Wu, A. Souza, T. Zhang, C. Fifty, T. Yu, and K. Weinberger, "Simplifying graph convolutional networks," in *International conference on machine learning*. Pmlr, 2019, pp. 6861–6871.