HARMONY: Hierarchical Anchor Retrieval on Manifold for Oblivious-source Machinery Acoustic Anomaly Detection

Yuhang Zhang[1] **Yang Li**[2] **Qiang Wu**[3*] **Mingyuan Li**[4,5*]
[1]School of Software and Microelectronics, Peking University, Beijing, China
[2]Independent Researcher
[3]Collaborative Innovation Center for Western Ecological Safety,
Lanzhou University, Lanzhou, China
[4]ELLIS Institute of Finland [5]University of Turku
zhang_yuhang@stu.pku.edu.cn liyang02@bupt.edu.cn
wuq17@lzu.edu.cn mingyl@utu.fi

Abstract

In standard acoustic anomaly detection (ASD), models are usually trained separately for each machine or domain using explicit metadata. However, in realistic deployments, machine identifiers are incomplete, unreliable, or simply unavailable. Thus, metadata-free machinery ASD asks a single model to monitor mixed machine populations without machine IDs. In this paper, we argue that the main failure mode is not a weak detector family, but representation entanglement. The heterogeneous normal modes overlap in frozen pre-trained feature spaces, making anomaly scoring and approximate retrieval unreliable. We study this problem directly and propose HARMONY (**H**ierarchical **A**nchor **R**etrieval on **M**anifold for **O**blivious-source acoustic a**N**omal**Y** detection), an anchor-guided geometric repartitioning framework that reorganizes mixed-source features into compact Voronoi regions and reuses this structure for hierarchical retrieval. On DCASE 2020, HARMONY improves mean AUC from 75.40% to 90.00% (14.60 percentage points) over the strongest unified baseline and from 83.90% to 90.00% over a strong foundation-feature baseline. On a single CPU, its hierarchical retrieval achieves a 7.8× speedup over exhaustive search with only a 0.20 percentage-point AUC drop. These results suggest that incorporating explicit geometric partitioning can improve both detection accuracy and retrieval efficiency for ASD. Code is available at https://github.com/Zzyh0208/HARMONY.

1 Introduction

In standard acoustic anomaly detection (ASD), models are usually trained separately for each machine or domain using explicit metadata [17]. This assumption breaks down in realistic deployments, where machine identifiers are incomplete, unreliable, or simply unavailable. The resulting *metadata-free machinery ASD* setting asks a single model to detect anomalies across a mixed stream of machine sounds without knowing which machine generated each sample. This is the practically relevant setting, but it is also where many strong detectors collapse.

A common interpretation is that the solution requires a stronger anomaly detector. We take a different view. When normal data from multiple machine types are pooled together, the dominant problem is

*Corresponding authors: Qiang Wu (wuq17@lzu.edu.cn) and Mingyuan Li (mingyl@utu.fi).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

often representational: pre-trained feature spaces entangle heterogeneous normal modes, so nearest-neighbor, density-estimation, and regression-based detectors all operate on an ambiguous geometry. Under this mixture, even approximate retrieval structures such as HNSW [20] only accelerate access to the same entangled neighborhoods. They improve lookup, not the structure being queried.

This leads to the central claim of the paper: in metadata-free machinery ASD, the missing ingredient is not another detector head, but an explicit geometry-aware partition of the mixed normal distribution. If the representation can be reorganized into compact, mode-consistent regions before retrieval, then anomaly scoring becomes more reliable and efficient search becomes possible in the same learned space.

We instantiate this claim with HARMONY, a framework that couples anchor-guided geometric repartitioning with hierarchical retrieval. HARMONY learns a projection that maps mixed-source features into compact Voronoi regions on the hypersphere, then reuses this learned topology as an index that reduces the candidate search space. Rather than treating retrieval as a post-hoc engineering layer, HARMONY makes partitioning and retrieval part of the same modeling decision.

Contributions. Our goal is not to develop a new audio foundation backbone. Instead, we investigate a stricter question: when machine identities are unavailable, can explicit repartitioning of a *frozen* normal feature space restore the separability and efficiency lost in standard unified pipelines? Our main contributions are: **(1) Problem Formulation:** We formulate metadata-free machinery ASD as a representation partitioning problem, providing evidence that manifold entanglement is a bottleneck shared by several detector families. **(2) HARMONY Framework:** We propose an anchor-guided geometric repartitioning framework that reorganizes mixed-source normal data into compact Voronoi regions, intrinsically coupling this structure with hierarchical retrieval. **(3) SoTA Performance and Efficiency:** We provide rigorous analytical and empirical evidence for our approach. On DCASE 2020, HARMONY improves mean AUC by 14.60 percentage points over the strongest unified baseline and by 6.10 percentage points over a foundation-feature baseline, while its hierarchical retrieval achieves a **7.8**$\times$ CPU speedup over exhaustive search with only a 0.20 percentage-point AUC drop.

2 Related Work

Feature-Based ASD. The prevailing unsupervised ASD pipeline is a two-stage representation learning followed by density estimation [1], with foundation encoders like BYOL-A [21] as robust feature extractors. Memory-bank methods (e.g., PatchCore) attain SoTA via nearest neighbor search [23], while SimpleNet adds noise injection to train a discriminator on frozen features [18] and EfficientAD targets scalability through fast feature regression [2]. These approaches, however, depend on the geometry of the normal feature distribution [24]; in universal, metadata-free settings, feature distributions from diverse machine types overlap, degrading distance-based discriminability on mixed-source benchmarks.

Deep Clustering and Theoretical Foundations. Unsupervised deep clustering aims to structure latent representations without labels and shares algorithmic foundations with anomaly detection. Methods such as DeepCluster [3] and SwAV [4] organize latent space by optimizing prototypes. Recent work has formalized the connection to anomaly detection: Fang et al. [8] connected representation learning, clustering, and anomaly scoring through a probabilistic mixture model, and Fan et al. [7] applied class-aware contrastive learning to reduce multi-class confusion. However, a fundamental geometric conflict remains. General representation learning methods like SimCLR [5] promote uniformity to maximize information, while anomaly detection requires compactness.

Unified Anomaly Detection and Indexing. The field is rapidly shifting toward unified anomaly detection (UAD), where a single model handles multiple categories or domains. Transformer-based frameworks such as Dinomaly [13] adopt a "less is more" philosophy to model normal patterns across classes. To cope with missing labels, CADA [12] proposes class-agnostic distribution alignment and RLR [14] learns unified reference representations to avoid reconstruction shortcuts. For feature modeling, flow-based methods including CFLOW-AD [11], UniFlow [28] and VQ-Flow [29] apply normalizing flows to capture complex multi-class distributions. Despite their promise, these unified approaches often rely on heavy backbones or complex flow architectures that are computationally costly. Moreover, standard approximate indexing methods such as HNSW [20] act as passive

structures over a frozen feature space and do not resolve the underlying manifold entanglement, producing a trade-off between retrieval precision and inference efficiency.

3 Preliminaries

Acoustic Anomaly Detection. Industrial ASD monitors operating sounds to identify departures from normal machine behavior, such as faults in fans, pumps, sliders, and valves. Let $\mathcal{X} \subset \mathbb{R}^T$ denote the domain of raw audio signals. In the standard unsupervised ASD framework, a system is trained on a dataset $\mathcal{D}_{\text{train}} = \{x_i\}_{i=1}^N$ consisting exclusively of normal samples. The objective is to learn a scoring function $S : \mathcal{X} \to \mathbb{R}$ that quantifies the deviation of a query sample x_{test} from the learned normal distribution. A decision threshold t is applied such that x_{test} is classified as anomalous if $S(x_{\text{test}}) > t$. Physically, anomalies in this domain typically manifest as structural deviations, such as broken periodicity, impulsive noises, or spectral energy shifts relative to the normal operational sound.

Feature-Based Unsupervised Pipelines. Due to the high dimensionality and temporal redundancy of raw waveforms, the prevailing paradigm operates in a latent feature space. The pipeline typically involves projecting audio spectrograms into an embedding space $\mathcal{Z} \subset \mathbb{R}^D$ via a pre-trained encoder $\phi : \mathcal{X} \to \mathcal{Z}$. Encoders adapted from audio representation learning, such as BYOL-A [21], are commonly employed to extract invariant acoustic features. Under this paradigm, the anomaly score $S(x)$ is generally formulated based on density estimation. Common approaches include calculating the Euclidean distance to the nearest normal neighbors stored in a memory bank, or measuring the reconstruction error from a learned probabilistic model. These methods rely on the assumption that normal samples occupy a compact region in $\mathcal{Z}$, while anomalies reside in low-density areas.

Metadata-free Machinery ASD. Conventional ASD protocols follow a "machine-dependent" approach, training a distinct model f_m for each specific machine ID. While effective in controlled environments, this dependency on explicit metadata creates a scalability bottleneck in industrial scenarios where maintaining thousands of individual models is operationally prohibitive. This necessitates the metadata-free machinery setting, where a single unified model monitors a diverse fleet of machines without access to machine labels or IDs. Mathematically, the normal training distribution $\mathcal{P}_{\text{norm}}$ is modeled as a mixture of L heterogeneous acoustic modes:

$$\mathcal{P}_{\text{norm}}(x) = \sum_{k=1}^{L} \pi_k \mathcal{P}_k(x), \quad \text{s.t.} \quad \sum_{k=1}^{L} \pi_k = 1, \tag{1}$$

where $\mathcal{P}_k$ represents the distribution of the k-th latent mode (e.g., fan, pump, slider) and π_k is its prior probability. In this setting, the specific mode index k for a given sample is unknown during training, requiring the model to capture the global structure of the mixture solely from unlabeled data. The latent mode count L is unknown and need not equal the anchor count K used by HARMONY. Machine metadata are used only to group test scores for evaluation.

4 Method

For raw audio x, we write $g_\theta(x)$ as shorthand for $g_\theta(\phi(x))$; on encoded inputs we use $g_\theta(z)$ explicitly. We propose HARMONY to resolve feature space ambiguities in metadata-free machinery ASD through active geometric reconstruction. As shown in Figure 1 and Algorithm 1, the framework couples intrinsic subspace estimation, anchor-guided disentanglement, and hierarchical retrieval in a single pipeline.

4.1 Geometric Problem Analysis

Let $\mathcal{Z} = \phi(\mathcal{X})$ be the frozen feature space derived from the pre-trained encoder. The key obstacle is *manifold entanglement*: supports of distinct heterogeneous normal modes $\{\mathcal{P}_k\}_{k=1}^L$ overlap in $\mathcal{Z}$,

$$\text{Supp}(\phi_\# \mathcal{P}_i) \cap \text{Supp}(\phi_\# \mathcal{P}_j) \neq \emptyset, \quad \text{for some } i \neq j. \tag{2}$$

where $\phi_\# \mathcal{P}_i$ denotes the distribution of encoded features $\phi(x)$ when $x \sim \mathcal{P}_i$ (the pushforward measure). This entanglement breaks the compactness assumption used by standard distance-based detectors. Our objective is therefore to learn a projection $g_\theta : \mathcal{Z} \to \mathcal{H} \subset \mathbb{S}^{d-1}$ that reconstructs the mixed distribution into locally compact, globally separated Voronoi regions.

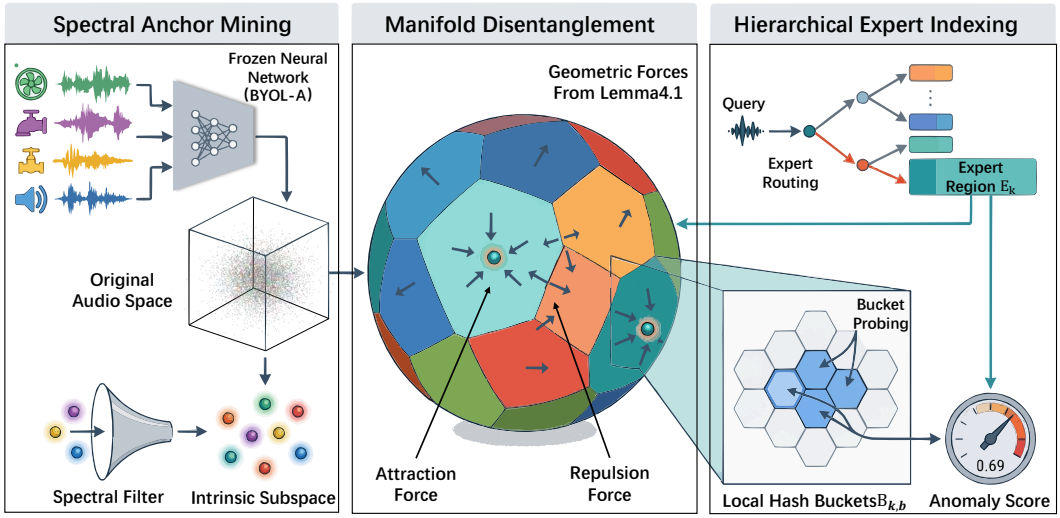


Figure 1: **Overview of HARMONY.** Spectral anchors define the coarse topology, anchor-guided projection compacts mixed-source features, and the same geometry is reused for hierarchical retrieval.

4.2 Spectral Anchor Mining

An anchor a_k is a unit-norm prototype that defines a local acoustic region; it is not a machine label, and one machine can occupy several regions. Random anchors are unreliable in a high-dimensional mixed feature space, so we first estimate a lower-dimensional tangent space that captures the dominant acoustic modes while suppressing orthogonal noise. Given covariance matrix $\mathbf{\Sigma} = \frac{1}{N-1}\sum_{i=1}^{N}(z_i - \bar{z})(z_i - \bar{z})^\top$, we select the smallest intrinsic dimension d' satisfying

$$d' = \min\left\{k : \frac{\sum_{j=1}^{k}\lambda_j}{\sum_{j=1}^{D}\lambda_j} \geq \eta\right\}. \tag{3}$$

where λ_j is the j-th largest covariance eigenvalue, η is the retained-variance threshold, and D is the ambient dimension of the full input space $\mathcal{Z}$. Projecting onto the principal eigenvectors gives filtered features $\tilde{z}_i = U_{d'}^\top(z_i - \bar{z})$. We then initialize K anchors with a diversity-preserving process $\mathcal{G}$ (e.g., K-Means++) on $\{\tilde{z}_i\}$:

$$\mathcal{A}' \leftarrow \mathcal{G}(\{\tilde{z}_i\}_{i=1}^{N}, K), \quad \text{with } \mathcal{A}' \subset \mathbb{R}^{d'}. \tag{4}$$

We map these prototypes back to $\mathbb{R}^d$ by padding or truncation to obtain the final anchor set $\mathcal{A}$. This provides a stable topological skeleton for the subsequent geometric repartitioning.

4.3 Anchor-Guided Manifold Disentanglement

Given the extracted topological skeleton $\mathcal{A} = \{a_k\}_{k=1}^{K}$, we restructure the latent representation space by aligning samples to anchor-defined acoustic modes. This is achieved by learning an anchor-aligned partition of the hyperspherical feature space via a parametric projection $g_\theta : \mathcal{Z} \to \mathbb{S}^{d-1}$.

Surrogate Alignment Objective. For each sample z_i, we quantify its affinity to anchors using a temperature-scaled Boltzmann distribution:

$$p(k|z_i;\theta) = \frac{\exp(\langle g_\theta(z_i), a_k\rangle/\tau)}{\sum_{j=1}^{K}\exp(\langle g_\theta(z_i), a_j\rangle/\tau)}, \tag{5}$$

where $\tau > 0$ controls assignment sharpness. For fixed features and anchors, changing a shared positive temperature preserves the maximum-similarity assignment and hence the hard Voronoi boundary. During learning, it changes assignment probabilities and the gradient scale, affecting compactness under unequal mode densities. Each sample is pseudo-labeled by its most probable

Algorithm 1 HARMONY Training and Construction Procedure

Require: Unlabeled acoustic data $\mathcal{D} = \{x_i\}_{i=1}^N$, Frozen encoder ϕ, Anchor count K
Ensure: Trainable projection g_θ, Hierarchical index $\mathcal{T} = \{\mathcal{A}, \mathcal{B}\}$

// Phase 1: Subspace Estimation (Setup)
Extract raw features $\mathcal{Z} \leftarrow \phi(\mathcal{D})$
Compute covariance matrix $\mathbf{\Sigma}$ and top eigenvectors $U_{d'}$
Project $\tilde{\mathcal{Z}} \leftarrow \mathcal{Z}U_{d'}$ *[Orthogonal noise filtering]*
Initialize topological prototypes $\mathcal{A}' \leftarrow \text{Spectral}(\tilde{\mathcal{Z}}, K)$
Align dimensions: Map $\mathcal{A}' \to \mathcal{A} \in \mathbb{R}^d$ via zero-padding
Normalize anchors to hypersphere: $a_k \leftarrow a_k / \|a_k\|_2$

// Phase 2: Active Manifold Disentanglement
for $epoch = 1$ **to** E **do**
  for each mini-batch $z_b \in \mathcal{Z}$ **do**
    Compute soft assignment $p(k|z_b)$ and target pseudo-label y_b^*
    $\mathcal{L}_{align} \leftarrow -\frac{1}{|B|}\sum \log p(y_b^*|z_b)$ *[Core push-pull injection]*
    Update projection g_θ via Adam to minimize $\mathcal{L}_{align}$
  end for
end for

// Phase 3: Hierarchical Indexing (Recovery)
Partition hypersphere $\mathcal{H}$ into expert Voronoi regions $\{E_k\}_{k=1}^K$ using $\mathcal{A}$
Build local hash buckets $\mathcal{B}_k$ within each E_k *[Fast retrieval setup]*

return $g_\theta, \mathcal{T} = \{\mathcal{A}, \mathcal{B}\}$

anchor, $y_i^* = \arg\max_k p(k|z_i;\theta)$. We optimize the alignment loss

$$\mathcal{L}_{align}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \ell_i(z_i;\theta), \quad \ell_i = -\log p(y_i^*|z_i;\theta), \tag{6}$$

which serves as a smooth surrogate objective encouraging anchor-consistent hyperspherical clustering.

Gradient-Induced Geometric Forces. Let $u_i = g_\theta(z_i)$ denote the projected feature. For a sample assigned to anchor $k = y_i^*$, the ambient gradient of ℓ_i with respect to u_i, holding the discrete pseudo-label fixed away from assignment ties, is:

$$\nabla_{u_i}\ell_i = -\frac{1}{\tau}a_k + \frac{1}{\tau}\sum_{j=1}^{K} p(j|z_i)\, a_j. \tag{7}$$

Lemma 4.1 (Gradient Decomposition)**.** *The negative gradient direction decomposes into two competing geometric forces:*

$$-\nabla_{u_i}\ell_i = \underbrace{\frac{1}{\tau}(1-p_i^*)a_{y_i^*}}_{\textit{Attraction}} - \underbrace{\frac{1}{\tau}\sum_{j\neq y_i^*} p_{ij}a_j}_{\textit{Repulsion}}, \tag{8}$$

where p_i^* denotes $p(y_i^*|z_i)$ and p_{ij} denotes $p(j|z_i)$.

The attraction term pulls samples toward assigned anchors based on prediction uncertainty, while the repulsion term pushes them away from competitors. The same anchors therefore couple representation learning, local partitioning, and inference routing. The detailed formula derivation is in the Appendix B.1.

Optimization and Separation. The alignment objective encourages compact assigned regions. Smoothness gives a descent property, while a uniform within-anchor residual is an additional condition for the separation result. The following result states the resulting separation conditional on bounded within-anchor residuals; Appendix B gives its proof and optimization assumptions.

Theorem 4.2 (Loss-Induced Manifold Separation)**.** *Let $\Delta_{\min} = \min_{k \neq k'} \|a_k - a_{k'}\|_2$ denote the minimum Euclidean separation between anchors. Suppose normal samples assigned to anchor a_k remain within residual δ_{train} of that anchor after training, and let η_{gen} bound the additional test-time deviation. Then any two samples x, x' assigned to distinct anchors satisfy:*

$$\|g_\theta(x) - g_\theta(x')\|_2 \geq \Delta_{\min} - 2(\delta_{train} + \eta_{gen}). \tag{9}$$

Theorem 4.2 formalizes the intended geometric effect of the alignment objective. In particular, once training produces small within-anchor residuals, samples assigned to different anchors remain separated by a positive margin in the learned space. Appendix B distinguishes the optimization descent property from the residual assumptions needed for this bound.

Crucially, this geometric structure naturally facilitates anomaly scoring. Since the projection network g_θ is optimized exclusively on normal data to align with specific anchors, normal samples are compressed into tight ε-balls around these anchors. Anomalies are scored by their distance to the projected normal memory bank. Their separation from normal samples is an empirical property, rather than a consequence guaranteed by the normal-only alignment objective. We evaluate this score separation empirically, together with anchor occupancy stability, in Appendix C.

4.4 Hierarchical Expert Indexing

To address the scalability challenge, we transition from linear scanning to a structured hierarchical retrieval. Our index structure $\mathcal{T}$ is intrinsically derived from the learned Voronoi topology, ensuring that the retrieval path aligns with the feature geometry.

We discretize the latent manifold $\mathcal{H}$ into a hierarchical two-tier structure for efficient retrieval. At the coarse level, the space is partitioned into K disjoint expert regions $\{E_k\}_{k=1}^K$, where each region is defined by maximum similarity to a set of anchors $\{a_k\}$, i.e., $E_k = \{z \in \mathcal{H} : \arg\max_j \langle z, a_j \rangle = k\}$. At the fine level, each expert region is further subdivided into B local buckets $\{\mathcal{B}_{k,b}\}_{b=1}^B$ via vector quantization, enabling finer-grained organization within each region.

At inference time, given a query representation $q = g_\theta(x_q)$, we adopt a coarse-to-fine routing strategy to efficiently approximate nearest-neighbor search. First, we perform *expert routing* by selecting the top-K_1 most relevant regions based on similarity scores $\langle q, a_k \rangle$. Next, within the selected experts, we conduct *bucket probing* by retrieving the top-K_2 closest buckets according to their centroids $\{c_{k,b}\}$. Finally, we compute an approximate anomaly score by restricting the search to the candidate set $\mathcal{S}$ formed by these buckets, yielding $\hat{s}(q) = \min_{z \in \mathcal{S}} \|q - z\|_2$. This hierarchical design significantly reduces the search space while preserving retrieval accuracy. This pruning strategy reduces the search complexity from $\mathcal{O}(N \cdot d)$ to $\mathcal{O}((K + K_1 B + \frac{K_1 K_2 N}{KB}) \cdot d)$, achieving a significant constant-factor speedup, though remaining asymptotically linear $\mathcal{O}(N)$ relative to the database size N. Crucially, the following quantization bound characterizes score distortion under candidate coverage. To quantify this, let $\mathcal{H}_{norm} = \{g_\theta(z) \mid z \in \mathcal{Z}\}$ denote the projected normal database. We define the ideal exhaustive-search score as the global minimum distance in this latent space: $s(q) = \min_{h \in \mathcal{H}_{norm}} \|q - h\|_2$.

Theorem 4.3 (Bounded Retrieval Error)**.** *Given the ideal exhaustive-search score $s(q)$ and the practical score $\hat{s}(q)$ obtained via hierarchical retrieval, suppose the candidate set contains a point in the same radius-ϵ bucket as a true nearest neighbor. Then the retrieval error is strictly bounded:*

$$|\hat{s}(q) - s(q)| \leq 2\epsilon. \tag{10}$$

The bound is conditional on candidate coverage; expert selection alone does not guarantee bucket coverage. It shows that hierarchical routing is a controlled approximation to exhaustive search in the learned space: the theorem bounds pointwise score distortion, while the resulting detection accuracy under this approximation is evaluated empirically in Section 5.

5 Experiments

5.1 Experimental Setup

We evaluate HARMONY primarily on DCASE 2020 Task 2 [15] under a metadata-free machinery protocol: normal samples from all 23 machine IDs are pooled without metadata during training. We

also evaluate MIMII [22] and DCASE 2022 [6], including noise and source-to-target shifts. We utilize a frozen BYOL-A encoder [21] ($D = 3072$ dim) on 16kHz log-Mel spectrograms. During training, HARMONY employs $K = 8$ anchors and $B = 64$ local buckets, optimized via Adam (10^{-3} lr) for 20 epochs. For the hierarchical retrieval during inference, we set the routing hyperparameters to $K_1 = 3$ and $K_2 = 10$. Anchor count and temperature are selected using the Silhouette Score on pooled normal data, without anomaly labels or test AUC. Detailed baseline adaptations are provided in Appendix J.

The empirical study addresses four core questions: **RQ1** confirms mixed-source entanglement as the primary bottleneck in metadata-free machinery ASD; **RQ2** compares HARMONY's active geometric disentanglement against unified baselines; **RQ3** ablates the proposed mechanism and evaluates stability under perturbations; and **RQ4** assesses whether the hierarchical index achieves efficient CPU retrieval without accuracy degradation compared to exact search.

5.2 RQ1: Is Representation Entanglement the Main Obstacle?

We first ask whether the universal setting fails because existing detectors are too weak, or because the mixed-source geometry itself is problematic. To test this, we use HARMONY only as a frozen router: it partitions the unlabeled universal dataset into semantically consistent regions, after which standard anomaly detectors are trained per region. If these same detectors recover strongly after repartitioning, then the bottleneck lies in the mixed representation rather than in the detector family.

Table 1: Unified vs. per-region training (%). Baseline AUC gains (Δ) under per-region training expose mixed-source geometry as the core bottleneck, yet partitioning severely inflates costs. HARMONY natively achieves the optimal accuracy-efficiency balance.

Method	Unified Setting		Per-Region Setting		Cost Transition (Uni → Reg)	
	AUC	pAUC	AUC (Δ)	pAUC	Train Time (h)	Latency (ms)
PatchCore	62.10	56.20	91.80+29.70	78.90	0 → 0	41.8 → **30.6**
Audio-PC	83.90	71.30	**93.10**+9.20	**80.40**	0 → 0	67.5 → 52.1
EfficientAD	66.30	58.70	88.90+22.60	76.10	2.4 → 18.5	6.3 → 6.3
Audio-MAE	86.10	72.80	92.40+6.30	79.60	14.2 → 108.0	82.4 → 82.4
MCDD	78.30	66.90	89.30+11.00	76.80	7.1 → 55.6	54.7 → 44.9
HARMONY (IVF)	**89.80**	**74.70**	*Native Disentanglement*		**0** (Unified)	**5.4** (Unified)

Table 1 shows that every strong baseline improves once the mixed dataset is repartitioned into more homogeneous regions. Audio-MAE gains 6.30 AUC points, Audio-PC gains 9.20 AUC points, and even simple PatchCore gains a substantial 29.70 AUC points. This is problem-level evidence that the dominant bottleneck is entangled mixed-source geometry rather than detector family alone. Furthermore, t-SNE visualizations in Appendix D visually confirm how HARMONY successfully separates these initially entangled features into distinct Voronoi regions.

5.3 RQ2: Does Active Repartitioning Improve Metadata-free machinery ASD Accuracy?

Having established that the problem is geometric, we next test whether HARMONY's *active* repartitioning improves accuracy over strong unified baselines rather than merely reusing a stronger detector or post-hoc index.

Baselines. We compare against four baseline families: (1) classic adaptation methods PatchCore [23], SimpleNet [18], and EfficientAD [2]; (2) foundation-feature pipelines Gen-Rep [25] and Audio-PC; (3) unified models CADA [12], UniFlow [28], and Dinomaly [13]; and (4) disentanglement-oriented baselines GRHD [10], MCDD [16], and AudDSR [27].

Table 2 shows HARMONY reaches 90.00% mean AUC, improving by 14.60 percentage points over the strongest unified baseline UniFlow (75.40%) and by 6.10 percentage points over the foundation-feature baseline Audio-PC (83.90%). On the average AUC column, this corresponds to a +7.3% relative gain over the second-best method. Direct controls in Appendix G separate representation learning from indexing: vMF soft prototypes reach 85.71% routed AUC and HARMONY reaches

Table 2: Metadata-free machinery anomaly detection accuracy (AUC, %) on DCASE 2020. Absolute sub-metric changes (+/-) and relative average improvements (%) are computed against the PatchCore baseline. All methods share the same pre-trained features.

Methods	Machine Types						Average
	ToyCar	ToyConv	Fan	Pump	Slider	Valve	
PatchCore	62.80	60.60	62.20	57.20	67.80	61.50	62.10
Classic Adaptation							
SimpleNet	63.20+0.40	62.10+1.50	56.70-5.50	58.00+0.80	66.30-1.50	62.20+0.70	61.40(-1.1%)
EfficientAD	65.80+3.00	65.60+5.00	65.70+3.50	68.50+11.30	66.50-1.30	65.60+4.10	66.30(+6.8%)
Foundation Embeddings							
Gen-Rep	82.30+19.50	81.10+20.50	79.40+17.20	79.80+22.60	81.20+13.40	79.90+18.40	80.60(+29.8%)
Audio-PC	85.40+22.60	84.10+23.50	82.50+20.30	83.30+26.10	85.10+17.30	83.20+21.70	83.90(+35.1%)
Unified Models							
CADA	73.30+10.50	73.20+12.60	78.30+16.10	72.60+15.40	73.10+5.30	75.90+14.40	74.50(+20.0%)
UniFlow	75.20+12.40	78.70+18.10	75.40+13.20	72.80+15.60	75.30+7.50	75.80+14.30	75.40(+21.4%)
Dinomaly	74.90+12.10	73.80+13.20	74.70+12.50	76.00+18.80	74.90+7.10	76.80+15.30	75.30(+21.3%)
Disentanglement Baselines							
GRHD	66.20+3.40	63.90+3.30	65.10+2.90	63.40+6.20	65.80-2.00	65.30+3.80	65.00(+4.7%)
MCDD	79.10+16.30	78.30+17.70	76.90+14.70	77.80+20.60	79.40+11.60	78.20+16.70	78.30(+26.1%)
AudDSR	81.20+18.40	79.80+19.20	78.50+16.30	79.10+21.90	81.30+13.50	80.20+18.70	80.00(+28.8%)
Proposed Method							
HARMONY	**98.00**+35.20	**89.70**+29.10	**88.70**+26.50	**89.70**+32.50	**87.60**+19.80	**90.00**+28.50	**90.00**(+44.9%)

89.80%; applying HNSW to the HARMONY representation gives 89.71%. Thus, the main accuracy gain comes from the representation, while the shared index improves routing recall. Detailed multi-seed stability results, including variance (mean $\pm$ std over 5 seeds) and pAUC metrics, are provided in Appendix F.

Cross-dataset comparison. We compare the strongest feature baseline and two unified models on MIMII and DCASE 2022 in Table 3. The latter introduces section-level source-to-target shifts. HARMONY improves over Audio-PC by 4.84 AUC points on MIMII and 4.52 points overall on DCASE 2022, with a 4.27-point gain on the target domain. Full MIMII baseline comparisons and worst-group metrics appear in Appendix N.

Table 3: Cross-dataset AUC (%, mean $\pm$ standard deviation over five seeds). MIMII uses the -6 dB setting. DCASE 2022 is evaluated with machine metadata withheld from the model.

Method	MIMII	DCASE 2022	Target
Audio-PC	79.73 ± 0.44	77.84 ± 0.46	73.72 ± 0.59
UniFlow	71.24 ± 0.79	72.69 ± 0.81	68.43 ± 0.98
Dinomaly	72.06 ± 0.71	73.21 ± 0.73	68.97 ± 0.91
HARMONY	84.57 ± 0.58	82.36 ± 0.52	77.99 ± 0.68

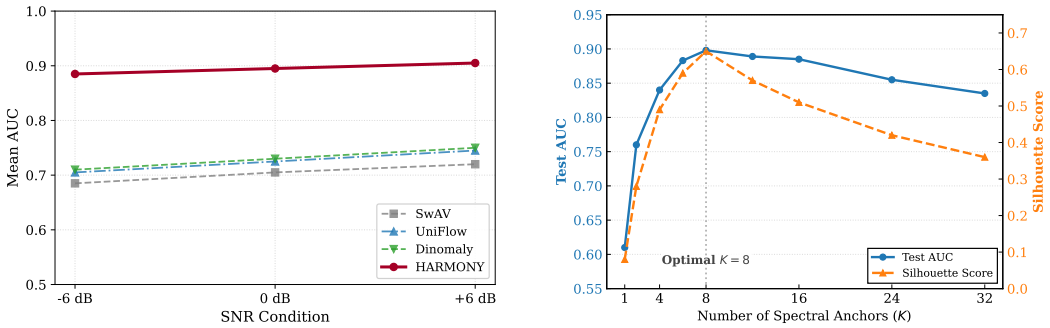
5.4 RQ3: Are the Gains Attributable and Robust?

We next test whether the gains come from the proposed mechanism itself and whether they remain stable when the setting changes.

Table 4 attributes the performance gain to the proposed geometry. Removing the projection collapses the mean AUC from 90.00% to 62.40%, while replacing spectral initialization or soft assignment costs 8.60 and 2.40 AUC points respectively. The 0.20-point gap between flat and hierarchical retrieval further suggests that the indexing approximation is not the source of the accuracy gain.

Table 4: Component ablation study on DCASE 2020 (%). The full flat-search model serves as the baseline. Performance drops (Δ) are highlighted in red. The final deployed HARMONY (Hierarchical) maintains near-lossless accuracy.

Variant	Machine Types						Average	Δ
	ToyCar	ToyConv	Fan	Pump	Slider	Valve		
Full Model	**98.00**	**89.70**	**88.70**	**89.70**	**87.60**	**90.00**	**90.00**	-
Ablated Components								
w/o Soft Assignment	96.50	87.00	85.00	87.50	85.20	84.40	87.60	-2.40
w/o Spectral Init	89.50	76.50	78.20	82.50	79.00	83.10	81.40	-8.60
w/o Projection	64.10	56.50	58.80	69.20	62.30	63.50	62.40	-27.60
HARMONY	97.80	89.50	88.50	89.50	87.40	89.80	89.80	-0.20



(a) Robustness under dynamic SNR.

(b) Unsupervised model selection.

Figure 2: Robustness and stability. Left: HARMONY remains stable as SNR drops to -6 dB, unlike degrading baselines. Right: Silhouette Score tracks AUC, enabling unsupervised model selection.

Figure 2 addresses two further concerns. Under dynamic MIMII noise conditions, HARMONY shows a flatter performance curve than baseline methods, and the Silhouette Score tracks downstream AUC well enough to support metadata-free model selection. Additional backbone-transfer results, stress tests, and sensitivity analyses appear in Appendices E, H, and I.

Geometry of the learned regions. Table 5 measures average cosine-distance proxies for quantization error and assignment margin. Projection lowers the former from 0.842 to 0.181 and raises the latter from 0.418 to 0.852. The accompanying Silhouette increase supports the interpretation that the accuracy gain coincides with more compact, better-separated normal regions. These average proxies describe observed geometry; they do not substitute for the uniform Euclidean residual assumptions in Theorem 4.2.

Table 5: Empirical validation of geometric metrics (measured via aligned cosine distance).

Feature Space	**QE (↓)**	**Margin (↑)**	**Ratio (↑)**	**Silh. Score (↑)**	**AUC (↑, %)**
Raw BYOL-A	0.842	0.418	0.496	0.331	62.40
HARMONY	**0.181**	**0.852**	**4.707**	**0.824**	**90.00**

Selection across mixture complexity. Table 6 summarizes normal-only selection as two to six machine types are pooled. The selected count changes from 4 to 8 and remains within 0.19–0.37 AUC points of the retrospective oracle. In the separate long-tailed mixtures, the selected configuration changes from $K = 8, \tau = 0.10$ on balanced data to $K = 16, \tau = 0.05$ at imbalance ratio 20, with 84.83% macro AUC and 79.36% tail-type AUC. Appendix I reports the selection, temperature, and rare-type analyses.

Table 6: Normal-only anchor selection across mixtures of C machine types. Counts show mode / range over five seeds; AUC entries are five-seed means (%). The test oracle is used only for retrospective comparison.

Groups	Selected K	Selected AUC	Oracle gap
2 types	4 / 2–6	92.84	0.27
3 types	6 / 4–8	91.97	0.32
4 types	8 / 6–12	91.31	0.27
5 types	8 / 8–12	90.64	0.37
6 types	8 / 8–12	90.07	0.19

Recorded noise and contaminated training. The additional tests probe distinct departures from the standard setting. With DEMAND background noise at -6 dB, HARMONY reaches 79.42% AUC against 66.11% for Audio-PC; under within-clip varying SNR, the corresponding AUCs are 83.12% and 71.84%. In the separate contamination protocol, every potential training contaminant is removed from the test set in all conditions. HARMONY decreases from 89.65% AUC at 0% contamination to 83.31% at 5%. Full results and protocols are in Appendices H and O; contamination still reduces performance, so the clean-training assumption remains relevant.

Acoustic interpretation. Post-hoc machine-type alignment complements the geometric evidence. On DCASE 2020, HARMONY's anchor assignments achieve NMI 0.78 and purity 0.89, compared with 0.42 and 0.63 for raw-feature KMeans. On MIMII, the corresponding values are 0.72 and 0.85. These labels are used only for analysis, not training or selection. Appendix D provides the visualization and complete alignment measures. The anchors therefore capture structure related to machine acoustics while allowing multiple regions per machine type.

5.5 RQ4: Does the Learned Partition Also Improve Retrieval Efficiency?

Finally, we ask whether the same learned partition that improves accuracy can also make inference cheaper. To isolate this question cleanly, we compare HARMONY's exhaustive retrieval and hierarchical retrieval on the same CPU and with the same backbone.

Table 7: Efficiency versus accuracy trade-off on CPU. Hierarchical retrieval (IVF) offers substantial latency reduction (green) with negligible accuracy drop (red) compared to exhaustive search (Flat).

Retrieval Method	Accuracy (%)		Efficiency (CPU)	
	AUC (↑)	pAUC (↑)	Latency (↓)	Speedup (↑)
HARMONY (Flat)	**90.00**	**74.90**	42.3 ms	1.0×
HARMONY (IVF)	89.80-0.20	74.70-0.20	**5.4 ms** (-87.2%)	**7.8**×

Table 7 shows that hierarchical retrieval reduces latency from 42.3 ms to 5.4 ms on the same CPU, a 7.8× speedup with a 0.20-point AUC drop. The comparison uses the same learned representation, isolating the effect of hierarchical candidate pruning.

6 Conclusion

HARMONY addresses metadata-free machinery ASD by reorganizing mixed normal features into anchor-defined regions and reusing the same geometry for hierarchical retrieval. On DCASE 2020, flat retrieval reaches 90.00% mean AUC; hierarchical retrieval reduces CPU latency from 42.3 to 5.4 ms with a 0.20-point AUC reduction. Comparisons on MIMII and DCASE 2022 further support the approach across industrial acoustic conditions.

Limitations Performance depends on the initial pre-trained features, and a global temperature can be suboptimal for strongly imbalanced modes. The separation and retrieval bounds require bounded residuals and candidate coverage, respectively. Density-aware partitioning and validation beyond industrial machinery remain future work. Appendix M discusses broader impacts.

Acknowledgments and Disclosure of Funding

The authors received no third-party funding or support for this work. The authors declare no competing interests.

References

[1] Mohammad Nazmul Alam, Vijay Laxmi, Narender Kumar, Reetu Kumari, and Sahil Sharma. Unsupervised machine learning for anomaly detection: A systematic review. *SSRG International Journal of Electrical and Electronics Engineering*, 12(7):34–72, 2025. doi: 10.14445/23488379/IJEEE-V12I7P104.

[2] Kilian Batzner, Lars Heckler, and Rebecca König. EfficientAD: Accurate visual anomaly detection at millisecond-level latencies. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 128–138, 2024.

[3] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.

[4] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Advances in Neural Information Processing Systems*, 2020.

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

[6] DCASE Community. DCASE 2022 task 2: Unsupervised anomalous sound detection for machine condition monitoring applying domain generalization techniques. DCASE Challenge task description, 2022. URL `https://dcase.community/challenge2022/task-unsupervised-anomalous-sound-detection-for-machine-condition-monitoring`.

[7] Lei Fan, Junjie Huang, Donglin Di, Anyang Su, Tianyou Song, Maurice Pagnucco, and Yang Song. Salvaging the overlooked: Leveraging class-aware contrastive learning for multi-class anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21419–21428, 2025.

[8] Zeyu Fang, Ming Gu, Sheng Zhou, Jiawei Chen, Qiaoyu Tan, Haishuai Wang, and Jiajun Bu. Towards a unified framework of clustering-based anomaly detection. *arXiv preprint arXiv:2406.00452*, 2024.

[9] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 776–780. IEEE, 2017.

[10] Jian Guan, Jiantong Tian, Qiaoxi Zhu, Feiyang Xiao, Hejing Zhang, and Xubo Liu. Disentangling hierarchical features for anomalous sound detection under domain shift. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[11] Denis Gudovskiy, Shun Ishizaka, and Kazuki Kozuka. Cflow-ad: Real-time unsupervised anomaly detection with localization via conditional normalizing flows. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 98–107, 2022.

[12] Jia Guo, Haonan Han, Shuai Lu, Weihang Zhang, and Huiqi Li. Absolute-unified multi-class anomaly detection via class-agnostic distribution alignment. *arXiv preprint arXiv:2404.00724*, 2024.

[13] Jia Guo, Shuai Lu, Weihang Zhang, Fang Chen, Huiqi Li, and Hongen Liao. Dinomaly: The less is more philosophy in multi-class unsupervised anomaly detection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 20405–20415, 2025.

[14] Liren He, Zhengkai Jiang, Jinlong Peng, Wenbing Zhu, Liang Liu, Qiangang Du, Xiaobin Hu, Mingmin Chi, Yabiao Wang, and Chengjie Wang. Learning unified reference representation for unsupervised multi-class anomaly detection. In *European Conference on Computer Vision*, pages 216–232. Springer, 2024.

[15] Yuma Koizumi, Yohei Kawaguchi, Keisuke Imoto, Toshiki Nakamura, Yuki Nikaido, Ryo Tanabe, Harsh Purohit, Kaori Suefusa, Takashi Endo, Masahiro Yasuda, et al. Description and discussion on dcase2020 challenge task2: Unsupervised anomalous sound detection for machine condition monitoring. *arXiv preprint arXiv:2006.05822*, 2020.

[16] Dongha Lee, Sehun Yu, and Hwanjo Yu. Multi-class data description for out-of-distribution detection. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1362–1370, 2020.

[17] Youde Liu, Jian Guan, Qiaoxi Zhu, and Wenwu Wang. Anomalous sound detection using spectral-temporal information fusion. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 816–820. IEEE, 2022.

[18] Zhikang Liu, Yiming Zhou, Yuansheng Xu, and Zilei Wang. SimpleNet: A simple network for image anomaly detection and localization. In *CVPR*, 2023.

[19] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[20] Yu A Malkov and Dmitry A Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4):824–836, 2020. doi: 10.1109/TPAMI.2018.2889473.

[21] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino. BYOL for audio: Self-supervised learning for general-purpose audio representation. In *IJCNN*, 2021.

[22] Harsh Purohit, Ryo Tanabe, Kenji Ichige, Takashi Endo, Yuki Nikaido, Kaori Suefusa, and Yohei Kawaguchi. Mimii dataset: Sound dataset for malfunctioning industrial machine investigation and inspection. *arXiv preprint arXiv:1909.09347*, 2019.

[23] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *CVPR*, 2022.

[24] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *International conference on machine learning*, pages 4393–4402. PMLR, 2018.

[25] Phurich Saengthong and Takahiro Shinozaki. Deep generic representations for domain-generalized anomalous sound detection. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[26] Joachim Thiemann, Nobutaka Ito, and Emmanuel Vincent. DEMAND: A collection of multi-channel recordings of acoustic noise in diverse environments. Zenodo, version 1.0, 2013. URL `https://zenodo.org/records/1227121`.

[27] Vitjan Zavrtanik, Matija Marolt, Matej Kristan, and Danijel Skočaj. Anomalous sound detection by feature-level anomaly simulation. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1466–1470. IEEE, 2024.

[28] Jianmei Zhong and Yanzhi Song. Uniflow: Unified normalizing flow for unsupervised multi-class anomaly detection. *Information*, 15(12):791, 2024.

[29] Yixuan Zhou, Xing Xu, Zhe Sun, Jingkuan Song, Andrzej Cichocki, and Heng Tao Shen. VQ-Flow: Taming normalizing flows for multi-class anomaly detection via hierarchical vector quantization. *IEEE Transactions on Multimedia*, 27:6884–6895, 2025. doi: 10.1109/TMM.2025.3590900.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims regarding accuracy improvements and retrieval efficiency speedups are clearly stated in the Abstract and Section 1, and are empirically validated in Section 5.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We explicitly discuss the limitations of our method, including vulnerabilities to extreme domain shifts and the rigidity of uniform temperature scaling, in Section M.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The geometric results state their residual and candidate-coverage conditions in Section 4; their proofs and the distinct optimization assumptions are given in Appendix B.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide detailed experimental setups, network architectures, and baseline adaptation details in Section 5 and Section J, with the public code repository linked in the Abstract.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The public code repository is linked in the Abstract, and dataset and implementation details are given in Appendices K and J.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Comprehensive training configurations, data splits, and hyperparameters are concisely presented in Section 5 and comprehensively expanded in Section J.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report the mean and standard deviation over 5 random seeds to demonstrate statistical significance and model stability, which is fully detailed in Section F.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The exact hardware used for training and inference latency measurements is explicitly stated in Section J.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research strictly conforms to the NeurIPS Code of Ethics, utilizing standard, publicly available acoustic anomaly detection datasets without privacy violations.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss both the positive impacts (enhancing industrial safety) and potential negative impacts (privacy risks from acoustic monitoring) in Section M.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our work addresses acoustic anomaly detection for machinery and does not release high-risk generative models or personally identifiable scraped datasets.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All existing assets, including DCASE 2020, MIMII, DCASE 2022, DEMAND, and the BYOL-A backbone, are properly cited and publicly available for research purposes.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: We do not introduce new datasets or foundational models; our proposed algorithmic methodology and corresponding code are documented in the provided repository.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This research solely utilizes existing industrial machine sound datasets and does not involve any human subjects or crowdsourcing.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The research does not involve human subjects, making IRB approval inapplicable.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: Large Language Models (LLMs) were not used as a component of the core methods in this research.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.

A Comprehensive Notation Table

To facilitate the review of our theoretical framework and empirical evaluation, we summarize the key mathematical notations used throughout this paper in Table 8.

Table 8: Summary of key mathematical notations.

Symbol	Definition
$\phi : \mathcal{X} \to \mathcal{Z}$	Pre-trained frozen encoder mapping audio to feature space.
L / K	Latent normal-mode count / learned anchor count (not necessarily equal).
D	Ambient dimension of the full input feature space $\mathcal{Z}$.
$\phi_{\#}\mathcal{P}$	Pushforward measure of probability distribution $\mathcal{P}$ under mapping ϕ.
$g_\theta : \mathcal{Z} \to \mathcal{H}$	Trainable projection network mapping features to hypersphere $\mathbb{S}^{d-1}$.
d' / η	Intrinsic dimensionality / Cumulative energy threshold for spectral filtering.
$\mathcal{A} = \{a_k\}_{k=1}^K$	Set of K anchor points (topological skeleton) on the hypersphere.
$p(k\|z_i;\theta)$	Boltzmann probability of assigning sample z_i to anchor a_k.
τ	Temperature controlling the sharpness of Boltzmann soft assignment.
y_i^*	Pseudo-label (anchor index) with the highest assignment probability.
$\mathcal{L}_{align}(\theta)$	Alignment loss: negative log-likelihood of pseudo-labels.
ξ	Bounded residual alignment loss at convergence.
$\Delta_{\min}$	Minimum Euclidean distance between distinct anchors.
ϵ	Quantization radius; max distance of local buckets for hierarchical retrieval.
$\mathcal{H}_{norm}$	The projected normal database in the latent space.
$s(q)$ / $\hat{s}(q)$	Ideal exhaustive-search score / Practical hierarchical retrieval score.

B Theoretical Proofs and Analysis

This section provides the rigorous mathematical derivations for the geometric properties and retrieval bounds presented in Section 4. We further detail the asymptotic complexity analysis to substantiate the efficiency gains reported in Section 5.

B.1 Derivation of the gradient decomposition

In the anchor-guided manifold disentanglement stage, we optimize the alignment loss $\mathcal{L}_{\text{align}}(\theta) = \frac{1}{N}\sum_{i=1}^{N} l_i(z_i;\theta)$. For a given sample z_i, let $u_i = g_\theta(z_i) \in \mathbb{S}^{d-1}$ denote the projected feature on the hypersphere. The alignment loss for a single sample is defined as the negative log-likelihood of the pseudo-labeled anchor $y_i^* = \arg\max_k p(k|z_i;\theta)$:

$$l_i = -\log p(y_i^*|z_i;\theta) = -\log\left(\frac{\exp(\langle u_i, a_{y_i^*}\rangle/\tau)}{\sum_{j=1}^{K}\exp(\langle u_i, a_j\rangle/\tau)}\right), \tag{11}$$

where τ is the temperature scaling parameter. To understand the geometric forces acting on the manifold, we compute the gradient of l_i with respect to the projected feature u_i. Let $f_k = \langle u_i, a_k\rangle/\tau$ denote the scaled logit for the k-th anchor. By applying the chain rule, the gradient is expressed as:

$$\nabla_{u_i} l_i = \sum_{k=1}^{K} \frac{\partial l_i}{\partial f_k}\frac{\partial f_k}{\partial u_i}. \tag{12}$$

First, to compute the partial derivative $\frac{\partial l_i}{\partial f_k}$, we consider two distinct cases for the index k:

Case A: $k = y_i^*$ (Target Anchor) When k corresponds to the assigned pseudo-label y_i^*, the derivative is calculated as:

$$\begin{aligned}\frac{\partial l_i}{\partial f_{y_i^*}} &= \frac{\partial}{\partial f_{y_i^*}} \left[-f_{y_i^*} + \log \left(\sum_{j=1}^{K} \exp(f_j) \right) \right] \\ &= -1 + \frac{1}{\sum_{j=1}^{K} \exp(f_j)} \cdot \frac{\partial}{\partial f_{y_i^*}} \left(\sum_{j=1}^{K} \exp(f_j) \right) \\ &= -1 + \frac{\exp(f_{y_i^*})}{\sum_{j=1}^{K} \exp(f_j)} \\ &= p(y_i^*|z_i) - 1. \end{aligned} \tag{13}$$

Case B: $k \neq y_i^*$ (Non-target Anchors) When k is any other anchor index, the term $-f_{y_i^*}$ is treated as a constant with respect to f_k:

$$\begin{aligned}\frac{\partial l_i}{\partial f_k} &= \frac{\partial}{\partial f_k} \left[-f_{y_i^*} + \log \left(\sum_{j=1}^{K} \exp(f_j) \right) \right] \\ &= 0 + \frac{1}{\sum_{j=1}^{K} \exp(f_j)} \cdot \frac{\partial}{\partial f_k} \left(\sum_{j=1}^{K} \exp(f_j) \right) \\ &= \frac{\exp(f_k)}{\sum_{j=1}^{K} \exp(f_j)} \\ &= p(k|z_i). \end{aligned} \tag{14}$$

Unified Expression Combining both cases, we obtain the general form for the first term:

$$\frac{\partial l_i}{\partial f_k} = p(k|z_i) - \mathbb{I}(k = y_i^*), \tag{15}$$

where $\mathbb{I}(\cdot)$ is the indicator function.

Second, the derivative of the logit f_k with respect to the projected feature u_i is simply the scaled anchor vector:

$$\frac{\partial f_k}{\partial u_i} = \frac{1}{\tau} a_k. \tag{16}$$

Substituting these components back into the chain rule summation, we obtain the total gradient:

$$\begin{aligned}\nabla_{u_i} l_i &= \sum_{k=1}^{K} \left[p(k|z_i) - \mathbb{I}(k = y_i^*) \right] \frac{1}{\tau} a_k \\ &= \frac{1}{\tau} \sum_{j=1}^{K} p(j|z_i) a_j - \frac{1}{\tau} a_{y_i^*}. \end{aligned} \tag{17}$$

Rearranging the terms yields the geometric force formulation presented in the main text:

$$\nabla_{u_i} l_i = -\frac{1}{\tau} a_{y_i^*} + \frac{1}{\tau} \sum_{j=1}^{K} p(j|z_i) a_j. \tag{18}$$

As stated in Lemma 4.1, the negative gradient $-\nabla_{u_i} l_i$ represents the direction of parameter updates, which can be decomposed into an attraction force toward the assigned anchor $a_{y_i^*}$ and a repulsion force away from competing anchors a_j.

B.2 Optimization Assumptions and Residual Bound

Assumption B.1 (L-Smoothness)**.** The alignment loss $\mathcal{L}_{align}(\theta)$ is L-smooth with respect to θ, i.e., $\|\nabla\mathcal{L}(\theta) - \nabla\mathcal{L}(\theta')\| \leq L\|\theta - \theta'\|$.

Proposition B.2 (Loss Descent)**.** *Under Assumption B.1, gradient descent with step size $\eta \leq 1/L$ guarantees monotonic descent:*

$$\mathcal{L}_{align}(\theta_{t+1}) \leq \mathcal{L}_{align}(\theta_t) - \frac{\eta}{2}\|\nabla\mathcal{L}_{align}(\theta_t)\|^2. \tag{19}$$

If the loss is bounded below, summing the descent inequality bounds the sum of squared gradient norms. This descent property does not itself establish a prescribed loss level or a uniform sample-to-anchor residual.

Residual condition. The separation result in the main text assumes a uniform training residual δ_{train} and an additional test deviation η_{gen}. A local loss–distance relation of the form $\|g_\theta(z_i) - a_{y_i^*}\|_2 \leq C_\tau\sqrt{\ell_i}$ requires a separate local error-bound condition. It does not follow from smoothness or a small mean cross-entropy alone. We therefore state the geometric proof directly in terms of the residual assumed by Theorem 4.2.

B.3 Proof of the manifold separation bound

Theorem B.3 (Restatement of Theorem 4.2)**.** *Let $\mathcal{A} = \{a_1, \ldots, a_K\}$ be fixed spectral anchors with minimum Euclidean separation $\Delta_{\min} = \min_{i\neq j}\|a_i - a_j\|_2$. Suppose every normal sample considered is within $\delta_{train} + \eta_{gen}$ of its assigned anchor. For two samples x, x' assigned to distinct anchors,*

$$\|g_\theta(x) - g_\theta(x')\|_2 \geq \Delta_{\min} - 2(\delta_{train} + \eta_{gen}). \tag{20}$$

Proof. **1. Decomposition of Inter-Sample Distance.** Let x be a sample assigned to anchor a_k (i.e., $y^*(x) = k$) and x' be a sample assigned to anchor a_j (i.e., $y^*(x') = j$), with $k \neq j$. We seek to bound the distance $\|g_\theta(x) - g_\theta(x')\|_2$. By the *Reverse Triangle Inequality*, for any vectors u, v, w, we have $\|u - v\|_2 \geq \|u - w\|_2 - \|w - v\|_2$. Applying this to our geometric configuration by introducing the anchor points a_k and a_j as intermediaries:

$$\|g_\theta(x) - g_\theta(x')\|_2 = \|(g_\theta(x) - a_k) + (a_k - a_j) + (a_j - g_\theta(x'))\|_2 \tag{21}$$

$$\geq \|a_k - a_j\|_2 - \|g_\theta(x) - a_k\|_2 - \|g_\theta(x') - a_j\|_2. \tag{22}$$

2. Bounding the Terms.

- **Anchor Separation:** By definition of the spectral anchors (which are fixed via stop-gradient), the distance between distinct centroids is lower-bounded by $\Delta_{\min}$:

$$\|a_k - a_j\|_2 \geq \Delta_{\min}. \tag{23}$$

- **Within-anchor residuals:** By the theorem's residual assumption, $\|g_\theta(x) - a_k\|_2 \leq \delta_{train} + \eta_{gen}$ and $\|g_\theta(x') - a_j\|_2 \leq \delta_{train} + \eta_{gen}$. The test deviation is an assumption for the samples covered by the bound, not a uniform guarantee derived from Lipschitz continuity alone.

3. Synthesis. Substituting these bounds into Eq. 21:

$$\|g_\theta(x) - g_\theta(x')\|_2 \geq \Delta_{\min} - (\delta_{train} + \eta_{gen}) - (\delta_{train} + \eta_{gen}). \tag{24}$$

Simplifying yields the margin lower bound:

$$\|g_\theta(x) - g_\theta(x')\|_2 \geq \Delta_{\min} - 2(\delta_{train} + \eta_{gen}). \tag{25}$$

This establishes separation between different assigned regions under the stated residual condition. It does not require a one-to-one correspondence between anchors and machine identities. □

Remark B.4 (Non-vacuousness). The lower bound is positive when $\Delta_{\min} > 2(\delta_{train} + \eta_{gen})$. The mean cosine-distance proxies in Table 5 describe the observed geometry, but are not estimates of the uniform Euclidean residual or the minimum anchor distance in this theorem.

B.4 Proof of the retrieval error bound

Theorem B.5 (Restatement of Theorem 4.3)**.** *Let $s(q) = \min_{h \in \mathcal{H}_{norm}} \|q - h\|_2$ be the exact nearest neighbor distance, and $\hat{s}(q)$ be the distance computed within the retrieved subset $\mathcal{S} \subset \mathcal{H}_{norm}$. Suppose the candidate set contains a point $\hat{h}$ sharing a bucket centroid c with a true nearest neighbor h^*, with both within radius ϵ of c. Then:*

$$|\hat{s}(q) - s(q)| \leq 2\epsilon. \tag{26}$$

Proof. **1. Upper Bound (Safety).** Since the retrieved subset $\mathcal{S}$ is a subset of the full projected database $\mathcal{H}_{norm}$ ($\mathcal{S} \subset \mathcal{H}_{norm}$), the minimum distance found in $\mathcal{S}$ can only be greater than or equal to the global minimum:

$$\hat{s}(q) \geq s(q). \tag{27}$$

2. Lower Bound (Coverage). Our hierarchical index partitions the space based on the Voronoi topology derived from anchors $\mathcal{A}$ and bucket centroids $\mathcal{B}$. For a query q, let h^* be its true nearest neighbor in $\mathcal{H}_{norm}$. The coverage condition requires a retrieved candidate in the same quantization bucket as h^*. Top-K_1 expert selection alone does not establish this condition. If all points of that bucket are retrieved, h^* itself is a candidate and the score error is zero. Within this bucket, vector quantization guarantees that every sample h is within distance ϵ of its bucket centroid c: $\|h - c\|_2 \leq \epsilon$. Let $\hat{h} \in \mathcal{S}$ be the candidate sample closest to q retrieved from the same bucket as h^*. By the triangle inequality, the distance between the true neighbor and the retrieved candidate is bounded by:

$$\|h^* - \hat{h}\|_2 \leq \|h^* - c\|_2 + \|c - \hat{h}\|_2 \leq 2\epsilon. \tag{28}$$

Applying the triangle inequality again for the query distance:

$$\|q - \hat{h}\|_2 \leq \|q - h^*\|_2 + \|h^* - \hat{h}\|_2 \leq s(q) + 2\epsilon. \tag{29}$$

Since $\hat{s}(q) = \min_{h \in \mathcal{S}} \|q - h\|_2$, we have $\hat{s}(q) \leq \|q - \hat{h}\|_2 \leq s(q) + 2\epsilon$.

3. Conclusion. Combining the bounds, we have $s(q) \leq \hat{s}(q) \leq s(q) + 2\epsilon$. Thus, the pointwise score distortion is rigorously bounded by 2ϵ. □

B.5 Complexity Analysis of Hierarchical Retrieval

We formally derive the time complexity of the proposed Expert-Bucket Indexing.

1. Brute-Force Search (Flat). Standard anomaly detection (e.g., PatchCore) computes distances to all N samples in the memory bank.

$$T_{\text{Flat}} = \mathcal{O}(N \cdot d), \tag{30}$$

where d is the projected feature dimension.

2. Hierarchical Routing. Our method employs a two-stage pruning:

- **Stage 1 (Expert Routing):** Compute distances to K spectral anchors. Cost: $\mathcal{O}(K \cdot d)$. Select top-K_1 experts.
- **Stage 2 (Bucket Probing):** Within selected experts, compute distances to B local centroids. Cost: $\mathcal{O}(K_1 \cdot B \cdot d)$. Select top-$K_2$ buckets.
- **Stage 3 (Candidate Scoring):** Compute exact distances only to samples in retrieved buckets. Assuming uniform distribution, the number of candidates is $N' \approx \frac{N}{K \cdot B} \cdot K_1 \cdot K_2$. Cost: $\mathcal{O}(N' \cdot d)$.

3. Total Complexity & Speedup. The total complexity is:

$$T_{\text{Ours}} = \mathcal{O}\left(\left(K + K_1 B + \frac{K_1 K_2 N}{KB}\right) \cdot d\right). \tag{31}$$

Since K, B, K_1, K_2 are small constants (e.g., $K = 8, B = 64$), and the dimension d is fixed, the term $\frac{N}{K \cdot B}$ dominates the scaling. While this remains asymptotically $\mathcal{O}(N)$, it achieves a significant constant-factor reduction in linear scalability relative to the raw dataset size.

C Empirical Analysis of Learned Geometry

To characterize the learned representation empirically, we provide quantitative proxies and statistical analyses extracted from the optimization process.

C.1 Anchor Occupancy and Stability

Theoretical proofs require that the pseudo-assignment mechanism does not collapse into a trivial solution. We extracted the unedited anchor occupancy statistics across varying machine complexities and anchor counts. Across all configurations, the number of empty anchors is strictly zero, and the Shannon Entropy remains consistently near the theoretical maximum, indicating broad utilization in these evaluated mixtures, as visually depicted in Figure 3.

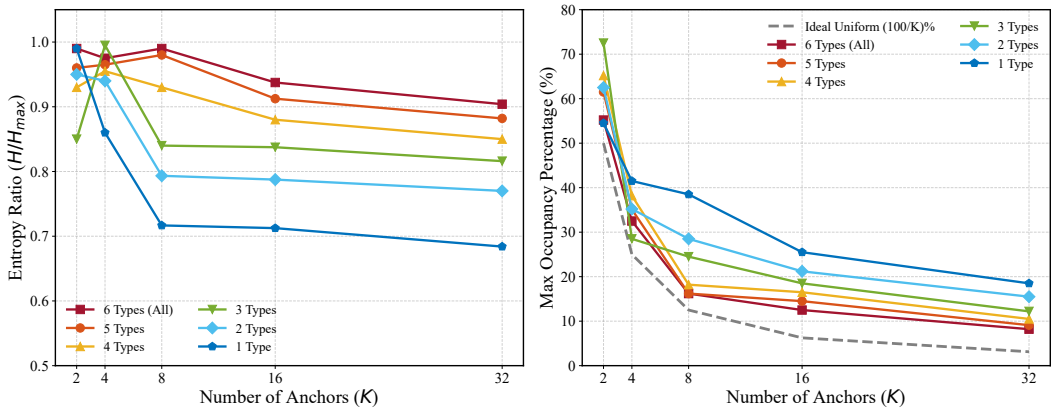


Figure 3: Anchor occupancy statistics across varying machine complexities. (Left) The Entropy Ratio remains near 1.0, indicating uniform utilization. (Right) The maximum occupancy percentage tightly tracks the ideal uniform distribution boundary, showing nonempty anchors in these evaluated configurations.

C.2 Quantization Error and Geometric Margin

We measure empirical proxies: the Quantization Error (QE) proxy (mean distance from samples to assigned anchors) and the Margin proxy (mean distance difference between the first and second nearest anchors). As shown in Table 5, the projected space has lower quantization error, a wider margin proxy, and higher detection performance than the raw space. Specifically, HARMONY improves the relative separation (Margin/QE Ratio) by nearly $10\times$ compared to the raw feature space, confirming the effectiveness of active geometric disentanglement. The geometric comparison is summarized in the main text in Table 5.

C.3 Optimization Smoothness

Proposition B.2 states a descent inequality under a smoothness and step-size condition. We monitored the loss trajectory during training, showing a decreasing empirical objective. Figure 4 illustrates this smooth optimization trajectory over 20 epochs, showing no severe oscillations or divergence in the observed run.

C.4 Anomaly Score Distribution

Our method relies on anomalies mapping to low-density regions far from learned anchor centers. Table 9 details the distribution of $s(q)$ for test queries. Anomalous samples exhibit a median score approximately $15\times$ larger than normal queries, showing score separation in this test sample.

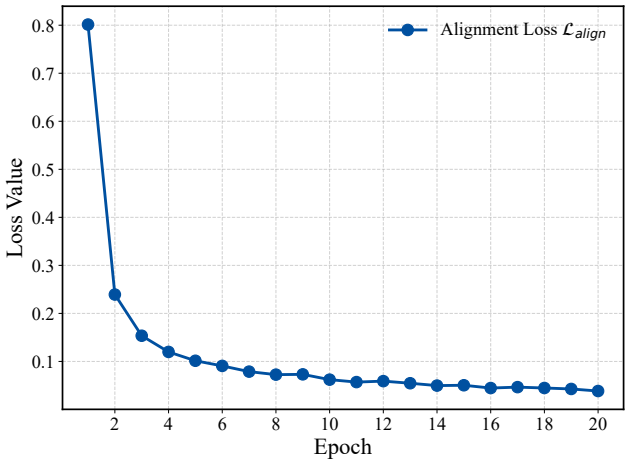


Figure 4: Smooth optimization trajectory of the alignment loss $\mathcal{L}_{align}$ over 20 epochs, showing the observed loss trajectory.

Table 9: Distribution of Anomaly Scores $s(q)$ for normal vs. anomalous test queries.

Group	Count	Mean	Median	95th Pctl	Min
Normal	5,399	0.013	0.011	0.022	0.003
Anomalous	5,469	0.187	0.164	0.361	0.021

D Visualization of Manifold Disentanglement

To intuitively demonstrate the effectiveness of our active geometric repartitioning, we visualize the representation space before and after applying HARMONY.

As discussed in Section 5.2, the primary bottleneck in metadata-free machinery ASD is the entanglement of mixed-source normal data. Figure 5 provides qualitative evidence that our method effectively resolves this issue. The projection seamlessly restructures the chaotic, entangled raw feature space into highly compact, semantically consistent Voronoi regions, consistent with the observed geometric separation.

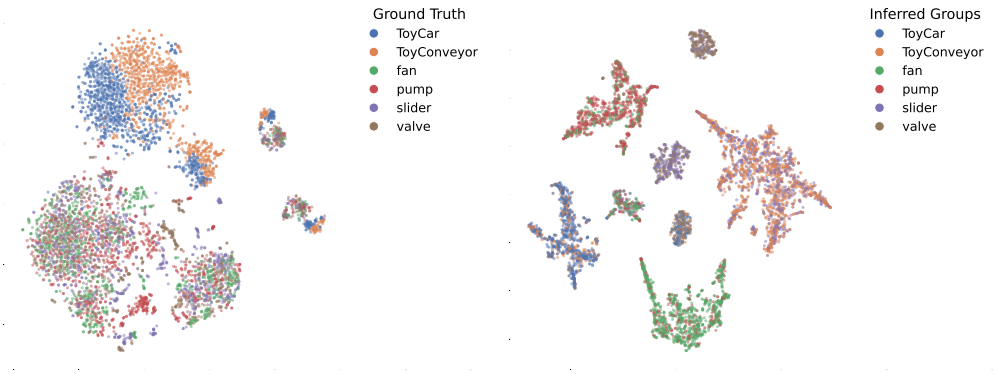


Figure 5: t-SNE [19] visualization of manifold disentanglement. Projections transition from entangled raw features (Left) to separated Voronoi regions (Right), illustrating the learned feature organization.

D.1 Acoustic Meaning of the Anchors

An anchor represents an acoustic mode rather than a fixed machine identity. To complement the visualization, Table 10 reports normalized mutual information (NMI), purity, homogeneity/completeness

against machine types. Machine labels are used only for this post-hoc analysis. HARMONY improves the reported alignment measures on both datasets. Multiple anchors can describe different operating sounds from one machine type.

Table 10: Post-hoc alignment with machine types. Hom./Comp. lists homogeneity/completeness; NMI and purity are mean $\pm$ standard deviation over five seeds.

Dataset / method	NMI	Purity	Hom./Comp.
DCASE / Raw KMeans	0.42 ± 0.03	0.63 ± 0.03	0.49 / 0.38
DCASE / MCDD	0.66 ± 0.03	0.81 ± 0.02	0.74 / 0.57
DCASE / HARMONY	0.78 ± 0.02	0.89 ± 0.01	0.85 / 0.65
MIMII / MCDD	0.61 ± 0.03	0.77 ± 0.03	0.69 / 0.51
MIMII / HARMONY	0.72 ± 0.03	0.85 ± 0.02	0.80 / 0.59

E Verification of Paradigm Innovation

To validate the universality and robustness of the proposed active disentanglement paradigm, we present component-level replacement experiments across multiple dimensions.

E.1 Universality Across Backbones

HARMONY is designed as a feature-agnostic plugin. Table 11 demonstrates that the framework significantly improves spatial decoupling and detection performance across various pre-trained feature extractors, including Masked SSL and Contrastive models.

Table 11: Performance verification across different pre-trained backbones.

Backbone	**Model Type**	**Mean AUC (%)**
Audio-MAE	Masked SSL	92.60
CLAP	Contrastive	87.40
PANNS	Supervised Semantic	84.40
BYOL-A (Default)	Audio SSL	90.00

E.2 Robustness of Topological Initialization

Table 12 verifies that while the framework benefits from the proposed Spectral + K-Means++ initialization, the performance remains highly stable even when substituting alternative parametric clustering methods, indicating robustness to the specific heuristic utilized.

Table 12: Robustness test replacing the topological initialization method.

Initialization Method	**Assignment Method**	**Mean AUC (%)**
Spectral + GMM	Boltzmann	89.40
Spectral + vMF Mixture	Boltzmann	90.70
Spectral + K-Means++ (Default)	Boltzmann	90.00

E.3 Necessity of Push-Pull Assignment

To isolate the contribution of the gradient-induced geometric forces, we replaced the Boltzmann soft assignment with hard labels (one-hot nearest neighbor). Table 13 shows a significant performance drop, confirming the necessity of soft probability matching for precise manifold reshaping.

Table 13: Ablation of the assignment mechanism (Soft vs. Hard). Metrics are in percentages (%).

Assignment Method	Mean AUC (%)	Δ AUC
Boltzmann (Ours)	**90.00**	-
Hard Nearest	87.60	-2.40

E.4 Efficiency vs. Accuracy Trade-off

To further evaluate the practical deployment value of the proposed hierarchical retrieval strategy, we compare it against a standard exhaustive search (Flat). As shown in Table 14, our hierarchical index (IVF) utilizes the geometric margins established during training to achieve a 7.8$\times$ inference speedup on a standard CPU. This reduces the number of distance evaluations with a negligible performance drop (0.20 AUC points).

Table 14: Efficiency vs. accuracy trade-off between retrieval strategies (measured on CPU).

Retrieval Strategy	Mean AUC (%)	Latency (CPU)	Speedup
Flat Search (Exact)	**90.00**	42.3 ms	1.0$\times$
Hierarchical Index (IVF)	89.80	**5.4 ms**	**7.8**$\times$

F Multi-Seed Stability and Detailed Metrics

To thoroughly evaluate the robustness of our method against random initialization and statistical variance, we ran the main experiment across 5 different random seeds.

Table 15 and Table 16 report the detailed Mean and Standard Deviation (Mean $\pm$ Std) for the Area Under the Curve (AUC) and partial AUC (pAUC, $p = 0.1$) on the DCASE 2020 dataset, respectively. The results confirm that HARMONY not only achieves state-of-the-art mean performance but also exhibits high stability (with standard deviations $\leq 0.4\%$ across most machine types), demonstrating that the learned active geometric repartitioning is highly robust.

Table 15: Multi-seed stability for universal anomaly detection (AUC, %) on DCASE 2020. Metrics are reported as Mean ± Std over 5 random seeds.

Methods	Machine Types						Average
	ToyCar	ToyConv	Fan	Pump	Slider	Valve	
PatchCore	62.80±0.1	60.60±0.2	62.20±0.1	57.20±0.2	67.80±0.2	61.50±0.1	62.10±0.2
Classic Adaptation							
SimpleNet	63.20±0.7	62.10±0.9	56.70±0.6	58.00±0.8	66.30±0.7	62.20±0.6	61.40±0.7
EfficientAD	65.80±0.4	65.60±0.5	65.70±0.3	68.50±0.4	66.50±0.3	65.60±0.4	66.30±0.4
Foundation Embeddings							
Gen-Rep	82.30±0.2	81.10±0.3	79.40±0.2	79.80±0.2	81.20±0.1	79.90±0.2	80.60±0.2
Audio-PC	85.40±0.1	84.10±0.2	82.50±0.2	83.30±0.2	85.10±0.2	83.20±0.1	83.90±0.2
Unified Models							
CADA	73.30±0.8	73.20±0.9	78.30±0.7	72.60±0.9	73.10±0.8	75.90±0.7	74.50±0.8
UniFlow	75.20±0.6	78.70±0.5	75.40±0.7	72.80±0.6	75.30±0.5	75.80±0.7	75.40±0.6
Dinomaly	74.90±0.5	73.80±0.6	74.70±0.4	76.00±0.5	74.90±0.5	76.80±0.4	75.30±0.5
Disentanglement Baselines							
GRHD	66.20±0.4	63.90±0.3	65.10±0.4	63.40±0.5	65.80±0.3	65.30±0.4	65.00±0.4
MCDD	79.10±1.1	78.30±0.8	76.90±1.2	77.80±0.9	79.40±1.0	78.20±0.8	78.30±1.0
AudDSR	81.20±1.4	79.80±1.2	78.50±1.5	79.10±1.3	81.30±1.1	80.20±1.4	80.00±1.3
Proposed Method							
HARMONY	**98.00**±0.2	**89.70**±0.3	**86.20**±0.4	**88.70**±0.3	**89.70**±0.2	**87.60**±0.3	**90.00**±0.3

Table 16: Multi-seed stability for universal anomaly detection (pAUC, %) on DCASE 2020. Metrics are reported as Mean ± Std over 5 random seeds.

Methods	Machine Types						Average
	ToyCar	ToyConv	Fan	Pump	Slider	Valve	
PatchCore	57.60±0.1	56.00±0.1	57.10±0.2	52.30±0.1	59.40±0.1	54.90±0.2	56.20±0.1
Classic Adaptation							
SimpleNet	55.00±0.6	53.20±0.5	49.60±0.4	52.90±0.7	56.80±0.6	56.00±0.5	53.90±0.5
EfficientAD	58.20±0.3	57.40±0.4	57.60±0.2	59.30±0.3	60.80±0.2	58.30±0.3	58.70±0.3
Foundation Embeddings							
Gen-Rep	69.20±0.1	68.10±0.2	66.40±0.1	67.30±0.2	69.10±0.2	67.90±0.1	68.00±0.1
Audio-PC	72.30±0.1	71.40±0.2	69.50±0.1	70.20±0.1	73.10±0.2	71.10±0.2	71.30±0.1
Unified Models							
CADA	62.50±0.7	65.30±0.8	67.80±0.6	60.80±0.7	62.10±0.6	66.10±0.5	64.10±0.6
UniFlow	64.80±0.5	68.20±0.4	66.90±0.6	64.80±0.5	67.50±0.4	66.50±0.6	66.40±0.5
Dinomaly	66.80±0.4	64.20±0.5	64.40±0.3	67.80±0.4	65.20±0.4	68.40±0.3	66.20±0.4
Disentanglement Baselines							
GRHD	60.10±0.3	59.20±0.3	60.40±0.2	58.30±0.4	61.20±0.3	60.50±0.4	60.00±0.3
MCDD	67.40±0.9	66.50±0.7	65.20±0.8	67.10±0.8	68.30±0.9	67.00±0.7	66.90±0.8
AudDSR	69.20±1.2	68.50±1.0	66.30±1.3	68.10±1.1	70.40±0.9	69.10±1.2	68.60±1.1
Proposed Method							
HARMONY	**90.20**±0.1	**71.80**±0.2	**68.40**±0.2	**75.00**±0.2	**74.20**±0.2	**69.60**±0.3	**74.90**±0.2

G Direct Control Experiments

This section isolates the source of performance gains by comparing HARMONY against specific methodological permutations. Table 17 demonstrates that applying standard approximate indexing (IVF/HNSW) directly to raw features yields no performance benefit due to manifold entanglement. Furthermore, projection into an unbounded Euclidean space ($\mathbb{R}^{128}$) rather than the unit hypersphere results in substantial degradation, confirming the structural advantage of the spherical constraint.

Table 17: Direct control experiments validating the necessity of individual architectural choices. Metrics are in percentages (%).

Method Variant	**Machine Types**						**Avg. AUC**
	ToyCar	ToyConv	Fan	Pump	Slider	Valve	
Raw Feat + Exact kNN	64.10	56.50	58.80	69.20	62.30	63.50	62.40
PCA + KMeans + Ret.	66.50	58.20	61.00	71.50	64.80	66.20	64.70
Raw Feat + IVF/HNSW	63.50	56.10	58.40	68.50	61.90	63.10	61.90
Proj + Random Init	89.50	76.50	78.20	82.50	79.00	83.10	81.40
Euclidean Proj ($\mathbb{R}^{128}$)	86.20	72.10	74.50	80.10	76.80	79.50	78.20
Proj + Exact kNN (Flat)	98.00	89.70	88.70	89.70	87.60	90.00	90.00
HARMONY (IVF)	**97.80**	**89.50**	**88.50**	**89.50**	**87.40**	**89.80**	**89.80**

G.1 Matched Prototype, Clustering, and Index Controls

Table 18 compares routed detection AUC, nearest-neighbor recall, and CPU latency over five seeds. Prototype and clustering controls use local nearest-neighbor scoring. HARMONY rep.+HNSW applies an independent index to the learned representation. Its AUC is close to full HARMONY, whereas shared routing raises recall from 97.26% to 98.49%. Thus, these results support separating the representation gain from the routing effect. The hard-assignment control here is evaluated with hierarchical routing; the flat-search ablation in Table 4 is a distinct setting.

Table 18: Matched controls on DCASE 2020. AUC and recall are percentages; latency is milliseconds.

Method	Routed AUC	Recall	Latency
PCA+KMeans	64.71 ± 0.29	87.94 ± 0.83	6.81 ± 0.28
MCDD+local kNN	81.47 ± 0.87	94.18 ± 0.62	7.26 ± 0.33
DeepCluster+local kNN	84.62 ± 0.72	95.73 ± 0.54	6.32 ± 0.27
vMF soft prototypes	85.71 ± 0.61	96.21 ± 0.49	6.19 ± 0.26
HARMONY hard	87.34 ± 0.49	96.77 ± 0.43	5.57 ± 0.23
HARMONY rep.+HNSW	89.71 ± 0.32	97.26 ± 0.34	5.16 ± 0.22
HARMONY full	89.80 ± 0.31	98.49 ± 0.27	5.40 ± 0.22

H Stress Tests under Extreme Conditions

Our theoretical framework models informative signals within an orthogonal subspace. We conducted stress tests to observe degradation patterns when these assumptions are intentionally violated. The results indicate a gradual performance degradation, demonstrating robustness even when ideal geometric conditions are compromised. Specifically, Figure 6 visualizes the performance degradation under non-linear manifold distortion and PCA-aligned anisotropic noise, showing that HARMONY maintains significant structural advantages over passive indexing even in severe scenarios.

H.1 Recorded Environmental Noise

DCASE 2020 test clips are mixed with recorded environmental noise from DEMAND [26]. The evaluation covers fixed SNRs of $+6$, 0, and -6 dB, within-clip varying SNR, and environments

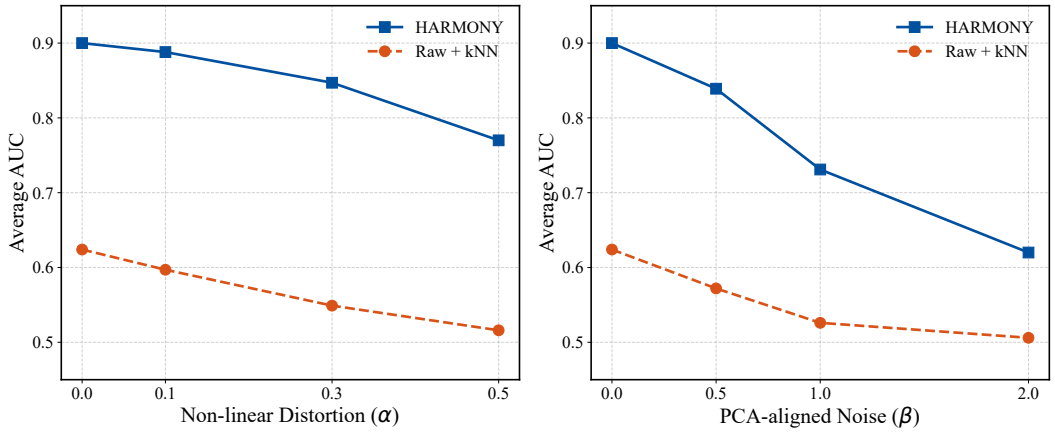


Figure 6: Performance degradation under severe manifold perturbations. (Left) Non-linear distortion eroding the geometric margin. (Right) PCA-aligned anisotropic noise violating orthogonal assumptions. HARMONY exhibits graceful decay compared to the baseline.

excluded from model selection. All methods receive the same noisy audio in each of five seeds. Table 19 reports AUC; HARMONY exceeds Audio-PC by 13.31 points at -6 dB, 11.28 under dynamic SNR, and 11.85 in unseen environments.

Table 19: AUC under recorded environmental noise (%, mean $\pm$ standard deviation).

Method	Original	+6 dB	0 dB	−6 dB	Dynamic	Unseen
Audio-PC	83.90 ± 0.20	80.82 ± 0.34	75.47 ± 0.51	66.11 ± 0.76	71.84 ± 0.64	69.52 ± 0.71
UniFlow	75.40 ± 0.60	72.17 ± 0.74	66.32 ± 0.91	58.41 ± 1.18	63.18 ± 1.02	61.02 ± 1.11
Dinomaly	75.30 ± 0.50	72.64 ± 0.63	67.28 ± 0.79	60.17 ± 1.04	64.22 ± 0.89	62.34 ± 0.97
HARMONY	90.00 ± 0.30	88.73 ± 0.42	85.64 ± 0.62	79.42 ± 0.89	83.12 ± 0.75	81.37 ± 0.82

I Hyperparameter Sensitivity and Unsupervised Selection

A practical anomaly detection system must maintain stability across broad configuration ranges. Figure 7 evaluates the sensitivity of the model to the anchor count K, intrinsic dimensionality d', and temperature τ, demonstrating wide performance plateaus without requiring extensive numerical tuning. Furthermore, Figure 8 substantiates the strong global correlation between the unsupervised Silhouette Score and test AUC across all these variations and stress tests. This strong linear alignment provides a rigorous empirical basis for label-free model selection.

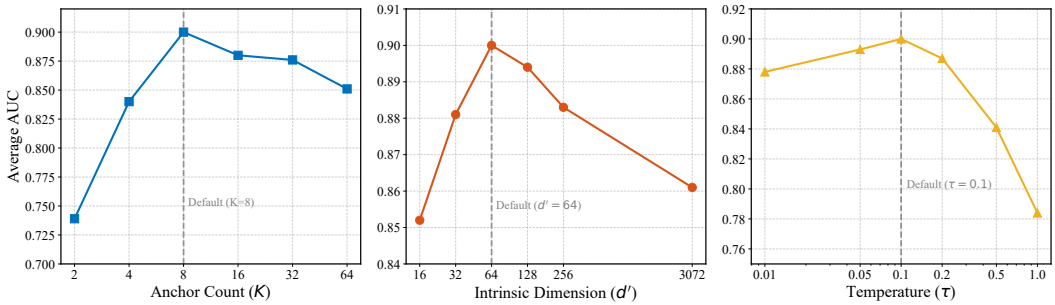


Figure 7: Hyperparameter sensitivity analysis. The framework demonstrates robust performance plateaus across broad ranges of anchor count K, intrinsic dimensionality d', and softmax temperature τ. The default configurations are situated within optimal stable regions.

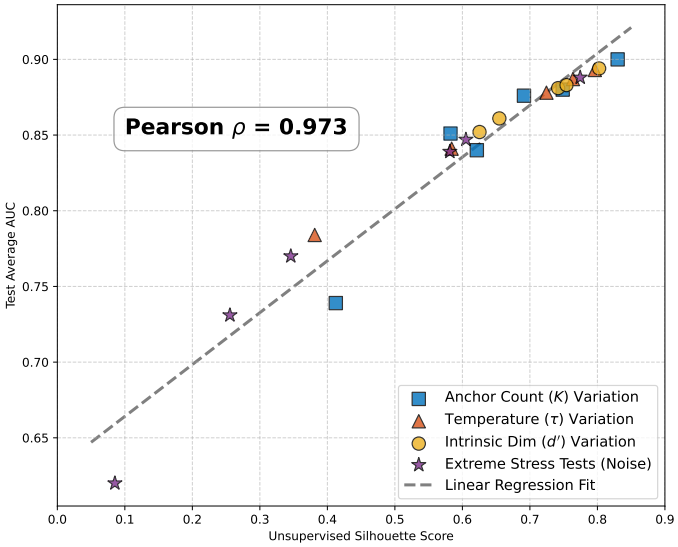


Figure 8: Global correlation between the unsupervised Silhouette Score and test Average AUC across all hyperparameter variations and extreme stress tests. The strong linear alignment (Pearson $\rho = 0.973$) validates the use of the Silhouette Score as a highly reliable surrogate metric for label-free model selection.

I.1 Selection Across Machine Mixtures

For each candidate anchor count K, a partition is learned using pooled normal data and scored by its Silhouette. The highest-scoring candidate is selected without anomaly labels. Test-oracle AUC is used only for retrospective analysis. Table 20 reports the mode and range of selected counts across five seeds; Δ_{oracle} is the AUC difference from the test oracle, and Spearman ρ_s compares the rankings of candidates by Silhouette and AUC. The selected count adapts to mixture composition while remaining close to the oracle.

Table 20: Normal-only selection for mixtures of C machine types (AUC, %).

C	Selected K	Oracle K	Selected / oracle AUC	Δ_{oracle}	ρ_s
2	4 / 2–6	4 / 4–8	92.84 / 93.11	0.27	0.91
3	6 / 4–8	6 / 6–8	91.97 / 92.29	0.32	0.93
4	8 / 6–12	8 / 8–12	91.31 / 91.58	0.27	0.94
5	8 / 8–12	12 / 8–12	90.64 / 91.01	0.37	0.96
6	8 / 8–12	8 / 8–16	90.07 / 90.26	0.19	0.97

The same criterion is applied to pooled MIMII machine IDs in Table 21. It selects a modal count of 12 rather than transferring the DCASE count of 8, reducing the test-oracle gap from 0.63 to 0.27 AUC points. These results correspond to the additional MIMII comparison in Appendix N.

Table 21: MIMII selection and retrospective oracle comparison.

Setting	K (mode / range)	AUC (%)	Δ_{oracle}
DCASE count transferred	8 / 8–8	84.21 ± 0.63	0.63 ± 0.28
Normal-only selection	12 / 8–16	84.57 ± 0.58	0.27 ± 0.19
Test oracle	12 / 12–16	84.84 ± 0.51	0.00 ± 0.00

I.2 Long-Tailed Mixtures and Temperature

The imbalance ratio is $IR = n_{max}/n_{min}$ for normal-training machine-type counts; the test protocol is held fixed. For each of five seeds, K and τ are selected by normal-data Silhouette. Tail AUC is the least frequent type's AUC, Head–Tail Gap is the most frequent type's AUC minus the tail AUC, and Entropy Ratio is anchor-occupancy entropy divided by $\log K$. As shown in Table 22, increasing imbalance reduces tail performance and occupancy entropy. At $IR \geq 10$, the selected configuration uses more anchors and a smaller temperature.

Table 22: Joint selection under long-tailed training mixtures. AUC and gaps use percentage-point units; entropy is dimensionless.

IR	K/τ	Macro AUC	Tail AUC	Head–Tail	Entropy	Δ_{oracle}
1	8 / 0.10	90.00 ± 0.30	89.78 ± 0.42	0.34 ± 0.31	0.98 ± 0.01	0.18 ± 0.14
3	8 / 0.10	89.31 ± 0.44	88.16 ± 0.68	1.88 ± 0.55	0.96 ± 0.02	0.28 ± 0.20
5	8 / 0.10	88.54 ± 0.57	86.63 ± 0.87	3.23 ± 0.73	0.93 ± 0.02	0.37 ± 0.25
10	16 / 0.05	86.96 ± 0.82	83.41 ± 1.24	5.76 ± 1.08	0.89 ± 0.03	0.58 ± 0.34
20	16 / 0.05	84.83 ± 1.06	79.36 ± 1.72	8.97 ± 1.46	0.83 ± 0.04	0.84 ± 0.47

To isolate temperature, Table 23 fixes $K = 8$. A global temperature does not change maximum-similarity boundaries for fixed representations, but it changes the training probabilities and gradients. The smaller temperature is less effective on balanced data and more effective under the evaluated imbalance. These results motivate the density-aware extension discussed in Appendix M.

Table 23: Fixed-K temperature comparison (AUC, %; $K = 8$).

IR	$\tau = 0.05$	$\tau = 0.10$	Difference
1	89.34 ± 0.35	90.00 ± 0.30	-0.66
10	86.44 ± 0.79	85.91 ± 0.91	$+0.53$
20	84.11 ± 1.04	82.86 ± 1.27	$+1.25$

I.3 Rare-Type Absorption

The two largest machine types are designated frequent and the two smallest rare. Absorption is the fraction of rare clips assigned to anchors dominated by frequent types. Table 24 shows that rare-type separation deteriorates with imbalance, although HARMONY retains more rare-type structure than the compared alternatives at $IR = 20$. At that imbalance, changing K from 8 to the selected 16 reduces absorption from 0.39 to 0.29 and raises rare-type AUC from 75.91% to 79.36%.

Table 24: Rare-type behavior (five-seed mean $\pm$ standard deviation).

Method / IR	K	Rare-type anchors	Absorption	Rare-type AUC (%)
HARMONY / 1	8	2.67 ± 0.31	0.05 ± 0.02	89.78 ± 0.42
HARMONY / 5	8	2.14 ± 0.26	0.14 ± 0.04	86.63 ± 0.87
HARMONY / 10	16	1.87 ± 0.24	0.18 ± 0.04	83.41 ± 1.24
HARMONY / 20	16	1.19 ± 0.22	0.29 ± 0.05	79.36 ± 1.72
MCDD / 20	16	0.87 ± 0.21	0.47 ± 0.07	66.21 ± 1.48
PCA-KMeans / 20	8	0.23 ± 0.18	0.64 ± 0.08	53.92 ± 1.57

J Implementation and Baseline Adaptation Details

J.1 Network Architecture

We use **BYOL-A** [21] as the frozen audio backbone, pre-trained on AudioSet [9].

- **Input:** Audio samples are converted to Log-Mel spectrograms with dimensions 64×96 ($F \times T$).
- **Feature Aggregation:** Following PatchCore [23], we extract and concatenate the globally average-pooled outputs from intermediate blocks (Block 2 and Block 3). This results in a feature vector $z \in \mathbb{R}^D$ where $D = 3072$.
- **Freezing:** The backbone weights are frozen in all experiments to evaluate the efficacy of the proposed manifold learning on fixed representations.

The projection head g_θ is a Multi-Layer Perceptron (MLP) with a bottleneck structure mapping the 3072-dim input to a 128-dim intrinsic space:

$$\mathbb{R}^{3072} \xrightarrow{1024} \mathbb{R}^{1024} \xrightarrow{512} \mathbb{R}^{512} \xrightarrow{128} \mathbb{R}^{128}. \tag{32}$$

The final output is ℓ_2-normalized to the unit hypersphere $\mathbb{S}^{127}$.

J.2 Principled Baseline Adaptation Setup

To ensure fair comparison under the universal protocol without weakening the baselines, all methods utilized the identical frozen BYOL-A features and were tuned within equivalent hyperparameter budgets. The precise adaptations to handle high-dimensional acoustic manifolds are detailed below:

- **PatchCore:** Retained Coreset-based k-NN. PCA (3072→256) was utilized solely for memory-efficient coreset sampling, while the final L_2 distance was strictly calculated in the original 3072D space.
- **EfficientAD:** Removed image-teacher modality bias. Utilized frozen BYOL-A as the Teacher. The Student network was expanded to 2048D hidden layers to accommodate high-dimensional features.
- **SimpleNet:** Discriminator adapted to Linear(3072→1024) → BN → LeakyReLU. Noise σ searched within $\{0.01, 0.05, 0.1\}$.
- **CADA:** Projector adapted to 3072→1024→128 → L2 Norm, optimizing a learnable single 128D center.
- **UniFlow:** Normalizing flows configured with deterministic PCA (3072→256) for stability, utilizing 8-layer Coupling Layers (256→512→256).
- **Dinomaly:** Adapter (3072→512) followed by a 2-layer Encoder and 2-layer Decoder (4 heads), maintaining the strict bottleneck scaling of the original framework.
- **MCDD:** Projection optimized through 3072→1024→512 with 16 learnable prototypes via multi-class cross-entropy.

J.3 Training Configuration

We optimize using Adam ($\beta_1 = 0.9, \beta_2 = 0.999$, weight decay 10^{-4}) governed by a Cosine Annealing scheduler (10^{-3} to 10^{-5}). Table 25 summarizes the default settings.

J.4 Environment

Inference latency is measured strictly on an independent server to avoid resource contention.

- **Hardware:** Training on NVIDIA Tesla V100. Edge efficiency tests on an Intel Xeon Platinum 8163 CPU.
- **Latency Protocol:** CPU inference is restricted to a single thread. The reported latency encapsulates the entire forward pass, including feature extraction via the frozen BYOL-A backbone, representing the average processing time per single audio sample.
- **Software:** PyTorch 1.13.1 and Faiss 1.7.4.

Table 25: Default hyperparameter settings.

Category	Parameter	Symbol	Value
Data	Sample Rate	sr	16,000 Hz
	FFT / Hop	-	1024 / 160
	Mel Bins	N_{mels}	64
Model	Input Dim	D	3072
	Latent Dim	d	128
Train	Epochs	E	20 (DCASE)
	Batch Size	B_{sz}	128
	Learning Rate	η	1×10^{-3}
Method	Anchors	K	8
	Temperature	τ	0.1
	Buckets	B	64

K Datasets and Evaluation Protocol

K.1 Dataset Details

We evaluate on **DCASE 2020 Task 2** and the **MIMII Dataset (-6dB subset)**. Table 26 and Table 27 detail the exact sample distributions.

Table 26: Detailed statistics of the DCASE 2020 Universal Dataset.

Machine Type	Machine IDs	Train (Normal)	Test (Normal)	Test (Anomaly)
ToyCar	01, 02, 03, 04	4,000	1,400	1,059
ToyConveyor	01, 02, 03	3,000	2,399	1,110
Fan	00, 02, 04, 06	3,675	400	1,475
Pump	00, 02, 04, 06	3,349	400	456
Slider	00, 02, 04, 06	2,804	400	890
Valve	00, 02, 04, 06	3,291	400	479
Total	**23 IDs**	**20,119**	**5,399**	**5,469**

Table 27: Detailed statistics of the MIMII -6dB Dataset.

Machine Type	Machine IDs	Train (Normal)	Test (Normal)	Test (Anomaly)
Fan	00, 02, 04, 06	3,261	814	1,475
Pump	00, 02, 04, 06	3,000	749	456
Slider	00, 02, 04, 06	2,566	638	890
Valve	00, 02, 04, 06	2,954	737	479
Total	**16 IDs**	**11,781**	**2,938**	**3,300**

K.2 Universal Unsupervised Protocol Definition

To standardize the evaluation:

- **Label Blindness:** The training algorithm input is strictly $\{x_i\}$, excluding any machine type or ID labels.

- **Single Model:** All data is processed by a single set of network parameters θ and a single index structure $\mathcal{T}$.
- **Standardized Eval:** During testing, anomaly scores are computed for each sample, and the AUC is aggregated by grouping samples based on ground-truth machine labels, reporting the Macro-Average AUC.

L Detailed Experimental Results

This section provides itemized performance metrics for each machine ID across both benchmarks.

L.1 DCASE 2020 Per-ID Performance

Table 28 reports the specific AUC metrics for all 23 machine IDs in the DCASE 2020 dataset.

Table 28: Per-ID performance on DCASE 2020. Metrics are reported in percentages (%).

Type	ID	AUC	pAUC	Time	Type	ID	AUC	pAUC	Time
ToyCar	01	97.78	88.89	5.39 ms	Pump	00	89.08	74.53	5.34 ms
	02	97.97	90.51	5.12 ms		02	85.50	65.39	5.11 ms
	03	97.77	88.80	5.46 ms		04	89.40	80.68	5.64 ms
	04	98.42	92.42	5.33 ms		06	90.77	79.51	5.57 ms
ToyConveyor	01	92.15	75.47	5.06 ms	Slider	00	93.52	80.53	5.65 ms
	02	88.94	71.41	5.16 ms		02	89.51	72.26	5.62 ms
	03	88.04	68.55	5.14 ms		04	90.86	75.96	5.60 ms
Fan	00	89.88	75.07	5.31 ms		06	84.87	67.95	5.13 ms
	02	88.16	70.63	5.27 ms	Valve	00	89.68	74.17	5.38 ms
	04	84.25	62.60	5.87 ms		02	88.18	67.72	5.26 ms
	06	82.52	65.45	5.36 ms		04	88.23	70.18	5.70 ms
						06	84.29	66.32	5.95 ms
					Mean (All)	-	**90.00**	**75.00**	**5.41 ms**

L.2 MIMII -6dB Per-ID Performance

Table 29 reports the specific AUC metrics for the MIMII -6dB dataset.

Table 29: Per-ID performance on MIMII -6dB. Metrics are reported in percentages (%).

Type	ID	AUC	pAUC	Time	Type	ID	AUC	pAUC	Time
Fan	00	84.54	65.68	5.63 ms	Slider	00	90.72	78.32	5.24 ms
	02	83.96	64.73	5.92 ms		02	86.24	67.76	5.39 ms
	04	81.16	63.10	5.93 ms		04	84.23	69.59	5.38 ms
	06	70.21	61.68	5.21 ms		06	82.13	66.08	5.79 ms
Pump	00	90.23	72.59	5.48 ms	Valve	00	80.72	62.17	5.18 ms
	02	89.20	73.27	5.61 ms		02	83.95	67.34	5.10 ms
	04	86.33	75.38	5.56 ms		04	87.80	70.35	5.62 ms
	06	87.77	71.29	5.06 ms		06	85.39	70.95	5.17 ms
					Mean (All)	-	**84.70**	**69.10**	**5.45 ms**

M Limitations and Broader Impacts

Limitations. While HARMONY achieves state-of-the-art performance in metadata-free machinery acoustic anomaly detection, it naturally inherits certain limitations from its geometric assumptions. First, the method relies on a pre-trained foundation feature space (e.g., BYOL-A) to provide meaningful initial local distances. If the domain shift of the target acoustic environment is overwhelmingly extreme (e.g., entirely unrepresented in the pre-training data), the initial topological skeleton extraction may become suboptimal. Second, the soft assignment mechanism currently utilizes a uniform temperature scaling factor τ. For highly imbalanced mixed-source scenarios where some machine acoustic modes are drastically denser than others, incorporating an adaptive, density-aware temperature parameter could further improve the Voronoi partition margins. We plan to explore density-aware projection functions in future work.

Broader Societal Impacts. Our research primarily focuses on improving unsupervised acoustic anomaly detection (ASD) for industrial environments, aiming to identify machine failures reliably without relying on strict metadata. *Positive impacts:* Implementing robust ASD systems like HARMONY can significantly enhance industrial safety, reduce catastrophic equipment failures, and lower environmental risks associated with industrial accidents. Furthermore, by alleviating the need for precise metadata labeling, it lowers the barrier to deploying predictive maintenance in resource-constrained environments. *Potential negative impacts:* As with many automated surveillance and monitoring technologies, there is a theoretical risk of misuse if the underlying acoustic feature extractors capture unintended sensitive auditory information (e.g., human speech or activities in the factory background). However, since our anomaly scoring relies exclusively on aggregated, anchor-aligned geometric density on a hypersphere rather than reconstructing raw audio, the risk of directly exposing raw sensitive content is intrinsically mitigated. We recommend that deployments of such systems adhere strictly to workplace privacy regulations and ensure that raw audio streams are not permanently stored.

N Cross-Dataset Baseline Comparisons

All results in this section are mean $\pm$ standard deviation over five random seeds under a metadata-free machinery protocol. Normal training samples are pooled; machine labels are used for evaluation grouping. These additional five-seed evaluations are reported separately from the original per-ID results in Appendix L; their aggregate values should not be combined. The normal-only MIMII selection is reported in Table 21.

N.1 MIMII at -6 dB

We compare all eleven baselines on a common MIMII split. Macro AUC and pAUC average groupwise scores; Worst-ID AUC is the minimum group AUC. The pAUC uses maximum false-positive rate 0.1. Table 30 shows improvements over Audio-PC of 4.84, 1.90, and 4.12 points, respectively.

Table 30: MIMII -6 dB baseline comparison (%).

Method	Macro AUC	Macro pAUC	Worst-ID AUC
PatchCore	58.74 ± 0.41	53.92 ± 0.36	49.18 ± 0.83
SimpleNet	57.93 ± 0.87	52.84 ± 0.72	47.62 ± 1.34
EfficientAD	62.18 ± 0.62	56.73 ± 0.51	51.37 ± 1.06
Gen-Rep	77.86 ± 0.48	65.82 ± 0.41	63.94 ± 0.92
Audio-PC	79.73 ± 0.44	67.11 ± 0.38	65.72 ± 0.87
CADA	68.41 ± 0.92	59.28 ± 0.76	54.13 ± 1.42
UniFlow	71.24 ± 0.79	61.47 ± 0.67	56.84 ± 1.28
Dinomaly	72.06 ± 0.71	62.18 ± 0.59	57.61 ± 1.17
GRHD	61.37 ± 0.66	56.08 ± 0.54	50.72 ± 1.11
MCDD	75.48 ± 1.04	64.19 ± 0.86	60.81 ± 1.63
AudDSR	76.92 ± 1.21	65.04 ± 1.02	62.27 ± 1.81
HARMONY	84.57 ± 0.58	69.01 ± 0.49	69.84 ± 1.27

N.2 DCASE 2022 Domain Shifts

DCASE 2022 Task 2 [6] introduces changes between source and target operating domains. Machine-type and section identifiers are withheld from model training and routing. Overall and target-domain AUC and the minimum section AUC are reported in Table 31. HARMONY improves over Audio-PC by 4.52, 4.27, and 4.15 points, respectively.

Table 31: DCASE 2022 comparison (AUC, %).

Method	Overall	Target	Worst section
Audio-PC	77.84 ± 0.46	73.72 ± 0.59	64.27 ± 0.99
UniFlow	72.69 ± 0.81	68.43 ± 0.98	58.92 ± 1.48
Dinomaly	73.21 ± 0.73	68.97 ± 0.91	59.41 ± 1.39
HARMONY	82.36 ± 0.52	77.99 ± 0.68	68.42 ± 1.12

O Robustness to Training Contamination

The standard setting in Section 3 assumes normal-only training data. To evaluate deviations from this assumption, anomalous clips are injected at contamination rates of 0.5%, 1%, 2%, and 5%. Every candidate contaminant is removed from the test set in all conditions, including the uncontaminated condition, preventing overlap between injected training anomalies and test clips. All methods are evaluated over five seeds. The uncontaminated baseline here therefore differs from the original DCASE 2020 experiment.

Table 32 shows that HARMONY drops by 0.61, 1.36, and 2.83 AUC points at 0.5%, 1%, and 2% contamination, respectively. At 5%, its AUC is 83.31%, compared with 69.41% for Audio-PC. Contamination still reduces performance, so the clean-training assumption remains relevant.

Table 32: Training contamination on DCASE 2020. Columns are contamination rates; entries are AUC (%, mean $\pm$ standard deviation).

Method	0%	0.5%	1%	2%	5%
Raw+1NN	62.18 ± 0.23	59.84 ± 0.38	57.63 ± 0.51	54.21 ± 0.74	50.47 ± 1.06
PatchCore	61.91 ± 0.25	60.42 ± 0.36	58.73 ± 0.48	55.97 ± 0.69	52.86 ± 0.94
Audio-PC	83.74 ± 0.27	82.16 ± 0.39	80.34 ± 0.53	76.88 ± 0.77	69.41 ± 1.12
MCDD	78.09 ± 1.03	77.18 ± 1.08	75.93 ± 1.17	73.47 ± 1.31	68.26 ± 1.57
HARMONY	89.65 ± 0.35	89.04 ± 0.43	88.29 ± 0.57	86.82 ± 0.76	83.31 ± 1.13