NOISE-AWARE UNSUPERVISED ANOMALOUS SOUND DETECTION WITH PRE-TRAINED AUDIO ENCODERS AND LORA ADAPTATION

Technical Report

Huan Yu[1]*, Yanjin Li*[1]*, Xuanting Fan*[2]*, Pan Li*[1]*, Zongmu Lin*[1]*,*
Zhongxin Bai[3]*, Gongping Huang*[1*]*,*

[1] Wuhan University, Wuhan, China,
[2] University of International Business and Economics, Beijing, China,
[3] Harbin Engineering University, Harbin, China,
yuhappy2007@outlook.com, gongpinghuang@whu.edu.cn

ABSTRACT

This technical report presents our submission to the DCASE 2026 Challenge Task 2. We adopt a two-stage framework consisting of a front-end feature extraction module and a back-end anomaly scoring module. Specifically, we extract representations using four pre-trained audio encoders—ATST-Frame, SSLAM, EAT-large, and BEATs—together with their LoRA-adapted variants. Features are extracted from multiple intermediate layers to enhance representational diversity. To improve robustness, the extracted features are further whitened before anomaly scoring. We investigate three complementary scoring strategies, including local-density k-nearest neighbors, relative Mahalanobis distance, and a hybrid GMM–cosine–kNN detector. The final anomaly score is obtained by first fusing scores across layers within each encoder and then aggregating them across different encoders. By ensembling different front-end representations and back-end scoring methods, we construct four systems for submission. On the development set, these systems achieve official scores of 66.65%, 65.58%, 64.02%, and 61.35%, respectively, consistently outperforming the official baseline of 57.66%.

Index Terms— Anomalous sound detection; Pre-trained audio encoders; LoRA adaptation; Ensemble anomaly scoring

1. INTRODUCTION

DCASE 2026 Challenge Task 2 [1, 2, 3, 4] extends the unsupervised learning, domain generalization, and first-shot settings from the 2023–2025 challenges. To improve resilience to background noise, it also introduces a new two-channel recording setup, where synchronized near- and far-field microphones are used to capture acoustic signals simultaneously.

The task targets four core challenges. First, models are trained exclusively on normal sounds and must learn compact representations of normality without access to anomalous data, despite highly diverse and unpredictable anomaly patterns. Second, models must remain robust for domain shifts induced by variations in machine operating conditions and environmental noise. Third, models must generalize to completely unseen machine types. Fourth, unlike previous settings, the task explicitly requires exploiting complementary information from synchronized near–far recordings to improve robustness to real-world industrial noise.

Our method is built upon three key strategies:

- First, we leverage the generalization capability of large-scale pre-trained audio encoders to satisfy the first-shot constraint.
- Second, to adapt to unseen machine types, we fine-tune lightweight LoRA modules using the additional training set.
- Third, to address domain shift between a data-rich source domain and a data-scarce target domain under a single threshold setting, we apply per-encoder feature whitening and fuse source-referenced and target-referenced anomaly scores.

2. SYSTEM DESCRIPTION

2.1. Overall framework

As illustrated in Fig. 1(a), the proposed framework is composed of a front-end feature extraction stage and a back-end anomaly scoring stage. The front-end aims to learn robust multi-layer acoustic representations. To this end, we employ four pre-trained audio encoders, together with LoRA-adapted variants of SSLAM, EAT-large, and BEATs, to enhance adaptation to unseen machine types. For each encoder, the most informative layer is selected for feature extraction. The back-end first converts the extracted representations into a whitened feature space. Anomaly scores are then computed using three complementary detectors, including Local Density KNN (LD-KNN), Relative Mahalanobis Distance (RMD), and a hybrid GMM–cosine–KNN model. Finally, score aggregation is performed in a hierarchical manner: scores are first averaged across selected layers within each encoder, and then aggregated across different encoders to produce the final anomaly score. Fig. 1(b) and Fig. 1(c) present the detailed structures of the front-end and back-end modules, respectively.

2.2. Front-end: robust feature extraction

We utilize four publicly available pre-trained audio encoders as feature extractors. For each encoder, we extract per-layer representations and aggregate them over time. Mean pooling is applied to ATST-frame [5], SSLAM [6] and EAT-large [7], while temporal pooling is used for BEATs [8].

*Corresponding author.

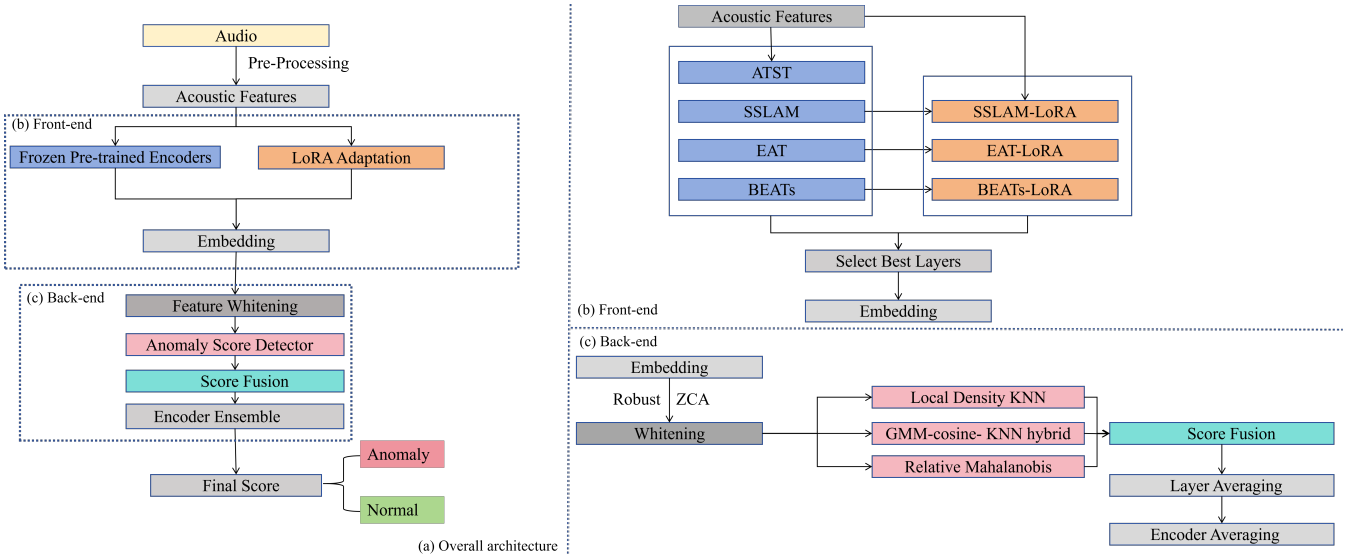

Figure 1: Overview of the proposed framework. (a) Overall architecture. (b) Front-end architecture. (c) Back-end architecture

For LoRA adaptation [9], all pre-trained parameters are frozen, and trainable low-rank matrices are inserted into attention projections. Specifically, give a pre-trained weight matrix $\mathbf{W} \in \mathbb{R}^{d\times k}$, the forward pass becomes

$$\mathbf{h} = \mathbf{W}\mathbf{x} + \Delta\mathbf{W}\mathbf{x} = \mathbf{W}\mathbf{x} + \mathbf{B}\mathbf{A}\mathbf{x}, \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{r\times k}$ and $\mathbf{B} \in \mathbb{R}^{d\times r}$ with rank $r \ll \min(d, k)$. Only $\mathbf{A}$ and $\mathbf{B}$ are trainable, enabling parameter-efficient adaptation. We apply such adapters to the attention projections of all Transformer blocks.

We explore two adaptation strategies: ArcFace-based supervised adaptation, and VICReg-based[10] self-supervised learning.

(i) ArcFace fine-tuning

For SSLAM and EAT-large, we adopt a *split Q/V* LoRA of rank 8 on the fused QKV projection, where only the query and value components are adapted, while the key projection remains frozen. For BEATs, we apply a *separate-Q/V* LoRA of rank 4 to these projections: attaches independent adapters to the individual `q_proj` and `v_proj` layers where Q, K, and V have their own weight matrices.

All LoRA parameters are optimized using the Sub-center ArcFace loss [11, 12] with 3 sub-centers, scale 30, and margin 0.2. The classification is defined over machine, domain, and attribute labels.

$$\mathcal{L}_{\text{ArcFace}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s\cdot\cos(\theta_{y_i}+m)}}{e^{s\cdot\cos(\theta_{y_i}+m)} + \sum_{j=1,j\neq y_i}^{n} e^{s\cdot\cos\theta_j}}, \tag{2}$$

where θ_{y_i} is the angle between the feature embedding and the ground-truth class center, and θ_j is the angle to the j-th class center. This is further regularized by a multi-layer MSE distillation loss between adapted and frozen features:

$$\mathcal{L}_{\text{distill}} = \frac{1}{L}\sum_{l=1}^{L}\|\mathbf{h}_l^{\text{adapted}} - \mathbf{h}_l^{\text{frozen}}\|_2^2, \tag{3}$$

Table 1: LoRA fine-tuning hyperparameters for ArcFace and VICReg.

Encoder	Method	LoRA type / rank	LR	Epochs	Batch
ArcFace fine-tuning on the additional training set					
SSLAM	ArcFace	split-QV / 8	1×10^{-4}	15	16
EAT-large	ArcFace	split-QV / 8	1×10^{-4}	15	8
BEATs	ArcFace	sep-QV / 4	1×10^{-5}	15	16
VICReg fine-tuning on the additional training set					
SSLAM	VICReg	full-QKV / 4	1×10^{-5}	20	20
EAT-large	VICReg	full-QKV / 4	1×10^{-5}	20	12
BEATs	VICReg	sep-QV / 4	5×10^{-5}	20	20

where l is the index of the layer. This constraint preserves the representational consistency of the frozen encoder and mitigates catastrophic forgetting.

To adapt to the five novel machine types in the evaluation set, we further fine-tune the LoRA adapters for 15 epochs on the additional training data, initializing from the checkpoints trained on the development set. The full hyperparameters are listed in Table 1.

(ii) VICReg fine-tuning.

For SSLAM and EAT-large, we apply a full-QKV LoRA of rank 4 with alpha 8 to all QKV and output projections. For BEATs, we apply a separate-Q/V LoRA of rank 4 with alpha 8 to the individual `q_proj` and `v_proj` layers: adds adapters to the entire fused QKV projection and the output projection of each attention block.

All adapters are trained with a VICReg self-supervised objective using invariance, variance, and covariance weights of 25, 25, and 1, respectively, with a machine-aware covariance regularization that computes the covariance loss within each machine type independently. All pre-trained encoder weights remain frozen, and only the LoRA adapters and a 3-layer MLP projection head are updated. Optimization uses AdamW with a cosine annealing schedule, a maximum learning rate of 10^{-5}, weight decay of 10^{-4}, and gradient clipping at 1.0, over 20 epochs. The hyperparameters are summarized in Table 1.

Table 2: Configuration of the four submitted systems.

System	Front-end	Back-end	Channel
System 1	ATST-Frame, SSLAM, EAT-large, BEATs, SSLAM-ArcFace, EAT-large-ArcFace, BEATs-ArcFace	LD + RMD fusion: $w_{\mathrm{LD}} = 0.7$, $w_{\mathrm{RMD}} = 0.3$, $w_{\mathrm{HB}} = 0$; z-normalization of scores; rank-normalized weighted averaging	Single
System 2	ATST-Frame, SSLAM, EAT-large, BEATs; SSLAM, EAT-large, BEATs trained with waveform perturbation; EAT-large-ArcFace, SSLAM-ArcFace, BEATs-ArcFace; and further fine-tuned on the "Additional training dataset".	LD + L2-normalized RMD fusion: $w_{\mathrm{LD}} = 0.7$, $w_{\mathrm{RMD}} = 0.3$, $w_{\mathrm{HB}} = 0$; rank-normalized weighted averaging	Single
System 3	SSLAM, EAT-large, BEATs, SSLAM-VICReg, EAT-large-VICReg, BEATs-VICReg	LD + RMD + GMM-cosine-KNN hybrid fusion: $w_{\mathrm{LD}} = 0.4$, $w_{\mathrm{RMD}} = 0.2$, $w_{\mathrm{HB}} = 0.4$; rank-normalized weighted averaging	Single
System 4	SSLAM, EAT-large, BEATs	LD + RMD + GMM-cosine-KNN hybrid fusion; $w_{\mathrm{LD}} = 0.25$, $w_{\mathrm{RMD}} = 0.35$, $w_{\mathrm{HB}} = 0.30$; LDN per branch;	Two

Table 3: Development-set results (AUC and pAUC, %). Baseline is the official Selective-Mahalanobis mode. "h-mean" is the per-axis harmonic mean over the seven machines; Ω is the official score, Eq. (10).

	Baseline			System 1			System 2			System 3			System 4		
Machine	AUC_s	AUC_t	pAUC	AUC_s	AUC_t	pAUC	AUC_s	AUC_t	pAUC	AUC_s	AUC_t	pAUC	AUC_s	AUC_t	pAUC
ToyCar	77.28	53.17	58.25	82.82	77.32	61.11	82.70	78.72	60.95	82.12	79.16	56.26	51.88	80.58	51.84
ToyCar(Emu)	69.49	66.62	53.47	69.42	77.54	52.21	69.52	77.76	51.68	65.90	85.02	52.79	66.34	69.78	51.79
bearing(Emu)	65.92	62.28	60.42	75.60	74.10	64.95	69.04	71.08	60.63	61.53	66.72	62.10	63.22	59.66	61.11
fan	60.00	45.09	52.29	76.98	71.18	53.89	73.50	67.66	52.58	58.20	63.34	52.37	53.32	52.88	51.05
gearbox(Emu)	74.48	52.74	53.97	80.26	76.16	54.58	78.60	76.12	55.00	69.89	68.96	53.47	70.70	67.42	54.26
slider(Emu)	66.36	49.18	50.36	57.34	66.64	52.79	56.74	66.20	52.74	68.76	69.06	53.95	66.32	63.28	55.16
valve(Emu)	56.60	56.50	50.20	74.56	84.14	52.84	75.68	84.06	52.26	72.42	81.44	52.68	76.10	82.62	67.89
h-mean	66.46	54.24	53.91	72.91	74.95	55.71	71.34	74.03	54.88	67.67	72.91	54.80	62.90	66.57	55.63
Ω		57.66			**66.65**			65.58			64.02			61.35	

2.3. Back-end: Whitened multi-layer anomaly scoring

Detection is performed for each encoder independently, and the resulting anomaly scores are subsequently aggregated across all encoders. For each encoder, a fixed set of intermediate transformer layers (Table 4) is selected. Given the extracted feature embeddings, the proposed back-end pipeline consists of three stages: feature whitening, complementary anomaly scoring, and score fusion.

- **Feature whitening [13].** To reduce the distribution discrepancy across different machine domains and improve the robustness of distance-based anomaly detection, the extracted features are first projected into a whitened feature space. The whitening transformation is estimated from the source-domain training features and subsequently applied to the source-domain, target-domain, and test features. Depending on the encoder, either Zero-Phase Component Analysis (ZCA) [14] whitening or robust singular-value [15] whitening is adopted. ZCA whitening is used for SSLAM and EAT-large encoders, while robust SVD whitening is applied to all BEATs encoders. Robust SVD whitening retains only the top principal components, explaining 97% of the total variance of the source-domain features. Other components are discarded to suppress noise amplification in the standard ZCA formulation. The whitening transformation is formulated as

$$\tilde{\mathbf{x}} = \mathbf{W}(\mathbf{x} - \boldsymbol{\mu}), \tag{4}$$

  where $\mathbf{x} \in \mathbb{R}^d$ denotes the feature embedding, $\boldsymbol{\mu}$ is the mean of the source-domain training features, and $\mathbf{W}$ denotes the whitening matrix.

- **Complementary anomaly scoring.** Three anomaly detectors are applied to the whitened features.

  - The Local Density (LD) [16] detector measures the local compactness of each sample with respect to its nearest neighbors, making it sensitive to local distribution changes.

$$\mathrm{LD}(\mathbf{x}) = \frac{\min_{\mathbf{r}\in\mathcal{R}} d(\mathbf{x}, \mathbf{r})}{\sum_{i=1}^{K} d(\mathbf{r}, \mathbf{r}_i)}, \tag{5}$$

    where $\mathcal{R}$ denotes the reference feature set, $d(\cdot,\cdot)$ denotes the Euclidean distance, and $\mathbf{r}_i$ represents the K nearest neighbors of $\mathbf{r}$. The score is computed independently on the source reference set K_s and the target reference set K_t.

  - The Relative Mahalanobis Distance [17] (RMD) detector captures the global statistical deviation by comparing the Mahalanobis distance to the class distribution and the global distribution. The RMD score is defined as

$$\mathrm{RMD}(\mathbf{x}) = D_M(\mathbf{x}, \boldsymbol{\mu}_c) - D_M(\mathbf{x}, \boldsymbol{\mu}_g), \tag{6}$$

    where $D_M(\cdot,\cdot)$ is the Mahalanobis distance, which is

$$D_M(\mathbf{x}, \boldsymbol{\mu}) = (\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}). \tag{7}$$

    where $\boldsymbol{\mu}_c$ and $\boldsymbol{\mu}_g$ denote the class-wise and global feature means, respectively, and $\boldsymbol{\Sigma}$ is the covariance matrix estimated from the whitened source-domain features.

Table 4: Selected feature layers and whitening strategies for each encoder, where layer indices start from 0.

Encoder	Whitening method	Selected layers
ATST-Frame	robust_whitening	1,0,2,8
SSLAM	zca_whitening	3,9,2
EAT-large	zca_whitening	12,3,21
SSLAM-LoRA	zca_whitening	3,4,2,10
EAT-large-LoRA	zca_whitening	11,13,4,3
BEATs	robust_whitening	6,3,4
BEATs-LoRA	robust_whitening	7,5,8,4

- The GMM-cosine-KNN hybrid method combines Gaussian mixture modeling, cosine similarity, and k-nearest neighbors. This hybrid score aims to assess both the probabilistic density and directional consistency of each sample, offering a complementary view that bridges local and global perspectives. Since the three detectors characterize different aspects of the feature space, they provide complementary anomaly evidence. The GMM-cosine-kNN hybrid score is defined as

$$\mathrm{HB}(\mathbf{x}) = w_{\mathrm{G}} \cdot s_{\mathrm{G}}(\mathbf{x}) + w_{\mathrm{c}} \cdot s_{\mathrm{c}}(\mathbf{x}) + w_{\mathrm{K}} \cdot s_{\mathrm{K}}(\mathbf{x}), \quad (8)$$

  where $s_{\mathrm{G}}(\mathbf{x})$ is the log-likelihood of $\mathbf{x}$ under a 4-component GMM[18] fitted to the whitened reference features, $s_{\mathrm{c}}(\mathbf{x})$ is the maximum cosine similarity between $\mathbf{x}$ and the reference embeddings, and $s_{\mathrm{K}}(\mathbf{x})$ is the z-normalized Euclidean distance from $\mathbf{x}$ to its nearest neighbor in the reference set, evaluated separately against the source and target references, and then combined via the minimum. The weights are set to $w_{\mathrm{G}} = 0.3$, $w_{\mathrm{c}} = 0.5$, and $w_{\mathrm{K}} = 0.2$.

- **Score fusion.** The outputs of the LD, RMD, and hybrid detectors are first normalized by z-score normalization and fused to produce a layer-level anomaly score. The resulting scores are then averaged across the selected layers to obtain an encoder-level score. Finally, the normalized scores of all encoders are averaged to generate the final anomaly score. The detector fusion is formulated as

$$S = w_{\mathrm{LD}} \cdot z(\mathrm{LD}) + w_{\mathrm{RMD}} \cdot z(\mathrm{RMD}) + w_{\mathrm{HB}} \cdot z(\mathrm{HB}), \quad (9)$$

  where $z(\cdot)$ denotes z-score normalization. HB denotes GMM-cosine-KNN hybrid detector.

Per-encoder scores are z-normalized and averaged with uniform weights. The binary decision threshold is the 90^{th} percentile of the source-domain training set scores, computed per machine.

3. EXPERIMENTS

3.1. Data preprocessing and implementation details

Models operate on the first channel of 10 s audio clips. SSLAM[1] and EAT-large[2] share the same input preprocessing, which extracts 128-bin log-mel spectrograms with a 10 ms shift, per-clip mean normalization, and padding or truncation to 1024 frames. In contrast, BEATs[3] and ATST-Frame[4] use their own native feature extractors. For ATST-Frame, SSLAM, and EAT-large, per-layer representations are mean-pooled over time, while BEATs applies its native temporal pooling. The optimization uses AdamW with a weight decay of 0.01, a cosine annealing learning rate schedule, gradient clipping at 1.0, and a class-balanced weighted sampler. Performance is evaluated separately for source and target domains using the area under the ROC curve, as well as the partial AUC over the low-false-positive range from 0 to 0.1. The final official score, denoted as Ω, is computed as the harmonic mean across machine types of the three individual quantities, namely source AUC, target AUC, and partial AUC, and then the harmonic mean is taken again across these three measures. The formula is given below:

$$\Omega = h\big(h_m \,\mathrm{AUC}_m^s,\ h_m \,\mathrm{AUC}_m^t,\ h_m \,\mathrm{pAUC}_m \big), \quad (10)$$

where h denotes the harmonic mean and h_m denotes the harmonic mean over machines.

3.2. Submitted systems and results

Table 2 summarizes the configurations of the submitted systems, including the front-end and back-end components as well as the employed audio channels.

As reported in Table 3, all four systems outperform the MAHALA baseline of 57.66% and the Simple-Autoencoder of 56.66% on the development set. System 1 uses ArcFace and additional data to train LoRA adapters, achieving the best $\Omega = 66.65\%$. System 2 extends the encoder set with additional LoRA variants trained with waveform perturbation, and adopts L2-RMD, reaching 65.58%, with the largest gains on the target domain. System 3 applies VICReg fine-tuning and keeps all three detectors LD, RMD, and GMM-cosine-KNN; it scores $\Omega = 64.02\%$ and a target-domain harmonic mean of 72.91%, on par with System 1. System 4 adopts two-channel input with LD, RMD, and cosine-KNN but no GMM, yielding 61.35%, still above baseline.

4. CONCLUSION

We presented a system for noise-aware first-shot unsupervised anomalous sound detection that pairs four frozen large audio encoders with rank-limited LoRA adapters and a training-free, whitened detector fusing a local-density k-NN distance and a relative Mahalanobis distance. The LoRA adapters are further fine-tuned on the additional training set to specialise in the evaluation of machine types. We submit an additional data-adapted ensemble and a broader generalist as complementary systems. Using only the near channel, the adapted system reaches an official score of 66.7% on the development set, against 57.7% for the official baseline. Exploiting the far-channel noise reference introduced this year is the natural next step.

5. REFERENCES

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

[2] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, M. Yasuda, and S. Saito, "ToyADMOS2: Another dataset of miniature-machine operating sounds for anomalous sound detection under domain shift conditions," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, Barcelona, Spain, November 2021, pp. 1–5.

[1] https://github.com/ta012/SSLAM/

[2] https://github.com/cwx-worst-one/EAT

[3] https://github.com/microsoft/unilm/tree/master/beats

[4] https://github.com/Audio-WestlakeU/audiossl/blob/main/audiossl/methods/atstframe

[3] K. Dohi, T. Nishida, H. Purohit, R. Tanabe, T. Endo, M. Yamamoto, Y. Nikaido, and Y. Kawaguchi, "MIMII DG: Sound dataset for malfunctioning industrial machine investigation and inspection for domain generalization task," in *Proceedings of the 7th Detection and Classification of Acoustic Scenes and Events 2022 Workshop (DCASE2022)*, Nancy, France, November 2022.

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

[5] N. Shao, X. Li, and X. Li, "Fine-tune the pretrained atst model for sound event detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2024, pp. 911–915.

[6] T. Alex, S. Atito, A. Mustafa, M. Awais, and P. Jackson, "Sslam: Enhancing self-supervised models with audio mixtures for polyphonic soundscapes," in *International Conference on Learning Representations*, vol. 2025, pp. 22 608–22 626.

[7] W. Chen, Y. Liang, Z. Ma, Z. Zheng, and X. Chen, "Eat: Self-supervised pre-training with efficient audio transformer," *arXiv preprint arXiv:2401.03497*, 2024.

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

[9] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, *et al.*, "Lora: Low-rank adaptation of large language models." vol. 1, no. 2, p. 3, 2022.

[10] A. Bardes, J. Ponce, and Y. LeCun, "Vicreg: Variance-invariance-covariance regularization for self-supervised learning," *arXiv preprint arXiv:2105.04906*, 2021.

[11] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4690–4699.

[12] J. Deng, J. Guo, T. Liu, M. Gong, and S. Zafeiriou, "Sub-center arcface: Boosting face recognition by large-scale noisy web faces," in *European Conference on Computer Vision*. Springer, 2020, pp. 741–757.

[13] S. Roy, A. Siarohin, E. Sangineto, S. R. Bulo, N. Sebe, and E. Ricci, "Unsupervised domain adaptation using feature-whitening and consensus loss," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 9471–9480.

[14] H. Li, X.-j. Wu, and T. S. Durrani, "Infrared and visible image fusion with resnet and zero-phase component analysis," *Infrared Physics & Technology*, vol. 102, p. 103039, 2019.

[15] W. Wang, Z. Dang, Y. Hu, P. Fua, and M. Salzmann, "Robust differentiable svd," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 9, pp. 5472–5487, 2021.

[16] R. G. Parr, S. R. Gadre, and L. J. Bartolotti, "Local density functional theory of atoms and molecules," *Proceedings of the National Academy of Sciences*, vol. 76, no. 6, pp. 2522–2526, 1979.

[17] J. Ren, S. Fort, J. Liu, A. G. Roy, S. Padhy, and B. Lakshminarayanan, "A simple fix to mahalanobis distance for improving near-ood detection," *arXiv preprint arXiv:2106.09022*, 2021.

[18] D. Roodman, "How to do xtabond2: An introduction to difference and system gmm in stata," *The stata journal*, vol. 9, no. 1, pp. 86–136, 2009.