HEADA: HYBRID EMBEDDING ADAPTATION FOR OPEN-SET RECOGNITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Adapting pretrained representations to downstream tasks is more challenging in an open world, where models often face unseen categories. In this setting, even strong pretrained embeddings (e.g., DINOv3 and SigLIP2) can remain under-adapted, as standard linear probing keeps the embedding space unchanged rather than adapting it for known–unknown separation. To reduce this gap, we propose Hybrid Embedding Adaptation (HEAda), a lightweight nonlinear projection head trained jointly with classification and metric-learning objectives. HEAda uses only known-class examples and operates directly on embeddings. By adapting both the embedding space and logits, HEAda supports wide range of open- and close-set methods within the same framework. HEAda consistently improves both open- and closed-set recognition over linear probing. It reduces mean AUROC error by 30%, 7%, and 18% for embedding-, logit-based, and hybrid OSR methods, respectively, while improving closed-set accuracy by 4.3 percentage points on average. But, more importantly, once the representation is properly adapted, simple KNN and Mahalanobis scores outperform more elaborate OSR methods, without additional detector training or auxiliary unknown data. This suggests that representation adaptation can matter more than the choice of OSR scoring rule.

1 INTRODUCTION

Modern pretrained models provide strong representations that transfer across a wide range of recognition tasks (e.g., image classification, retrieval, and fine-grained recognition) (Oquab et al., 2024; Siméoni et al., 2025; Tschannen et al., 2025; Stevens et al., 2024; Gu et al., 2025). Open-set recognition (OSR) puts additional demands on these representations: besides recognizing the classes seen during training, the model must reject samples from unseen classes (Scheirer et al., 2013; Yang et al., 2024; Li et al., 2025). Strong pretrained representations make it possible to approach this problem with little task-specific training, using the embeddings directly or fitting only a linear probe (Sun et al., 2022; Zhang et al., 2023; Adaloglou et al., 2023). **This raises a natural question: how much task-specific adaptation do strong pretrained representations still need for OSR?**

A standard lightweight approach to downstream adaptation is linear probing, where the pretrained embeddings are kept fixed, and only a task-specific classifier is learned. This provides lightweight adaptation to the known classes, but only at the classifier level; the embedding itself remains exactly as produced by the pretrained model. For OSR, this distinction matters because the embedding is not merely an input to the classifier. Many standard OSR methods operate directly on its geometry, for example through distances to class statistics or nearest training samples (Mensink et al., 2013; Lee et al., 2018; Sun et al., 2022), while others use the classifier outputs (Hendrycks & Gimpel, 2017; Liu et al., 2020). Linear probing therefore adapts one signal used for open-set recognition (i.e., the classifier outputs) while leaving the other (i.e., the embedding) unchanged.

Adapting the pretrained model can also reshape its representation, through full fine-tuning or more parameter-efficient approaches (Kumar et al., 2022; Houlsby et al., 2019; Hu et al., 2022; Jia et al., 2022). However, changing the model itself is not the only way to adapt its representation. Once the embeddings have been extracted, they can be transformed independently of the model that produced them. We therefore study a more restricted question: **can lightweight adaptation of the extracted embeddings improve OSR without modifying the pretrained model?**

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

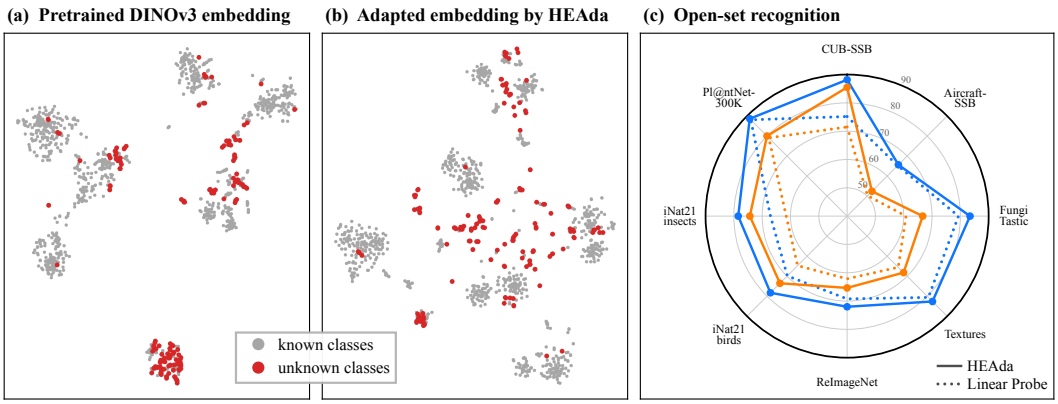


Figure 1: **HEAda makes pretrained features substantially stronger.** The reshaped feature space separates known and unknown class samples better, (a) and (b). Known class clusters are more compact, unknown samples tend to fall between them. This leads to consistent gains in both AUROC and OSCR. *Results are averaged over six encoders and, for open-set metrics, twelve post-hoc OSR methods. The t-SNE visualization of DINOv3-7B features on FungiTastic dataset.*

The answer is HEAda (Hybrid Embedding Adaptation), a lightweight nonlinear projection head that adapts already extracted embeddings for OSR. HEAda is trained using only known-class examples with a hybrid objective that combines classification and metric-learning supervision, producing both an adapted embedding and task-specific classifier logits. It requires no unknown or auxiliary outlier data and no access to the pretrained model during training. HEAda is not itself an OSR method: existing embedding-based, logit-based, and hybrid OSR methods are applied unchanged to its outputs, allowing us to isolate the effect of representation adaptation. We find that even strong pretrained representations remain under-adapted for OSR when used directly or with a linear probe. Across pretrained encoders, datasets, and OSR methods, HEAda improves embedding-based, logit-based, and hybrid OSR, while controlled comparisons show that these gains cannot be explained by increased head capacity alone. The same adaptation also improves closed-set recognition, rather than trading known-class accuracy for better unknown detection. But more surprisingly, after adaptation, simple distance-based scores such as KNN and Mahalanobis outperform more elaborate OSR methods. This suggests that, with strong pretrained features, adapting the representation may matter more than designing a more sophisticated open-set scoring rule.

Together, our study makes the following contributions:

1. **We identify a limitation of frozen pretrained features**, even with a linear probe, for open-set recognition. Across a wide range of pretrained encoders and datasets, we show that strong backbones (e.g., DINOv3) are under-adapted to separate samples from known and unknown classes, especially for fine-grained datasets, with considerable domain and distribution shifts.

2. **We address this gap with HEAda**, a multi-layer nonlinear projection head trained with a hybrid objective combining classification and metric-learning losses. HEAda operates directly on pretrained embeddings and adapts both the embeddings and logits for the downstream task. It requires no examples of unknown classes, no access to or backward pass through the encoder, and uses the same or fewer trainable parameters than a linear probe in our setting.

3. **We show that effective embedding adaptation requires more than additional model capacity.** The combination of nonlinear adaptation and metric-learning supervision is key to HEAda's gains, while deeper heads trained with classification alone provide only limited improvements.

4. **We demonstrate that these improvements are broad** and also carry over to closed-set recognition. HEAda reduces mean AUROC error by 30%, 7%, and 18% for common embedding-, logit-based, and hybrid OSR methods, respectively, and also improves closed-set mean accuracy across 78 encoder–dataset pairs by around 4 percentage points. Notably, after adaptation, simple KNN and Mahalanobis scores outperform the strongest OSR methods.

2 Related Work

Open-set recognition requires recognizing known classes and flagging unknown ones. Existing methods are primarily based on classifier outputs (e.g., confidence, logits, or energy) (Hendrycks & Gimpel, 2017; Hendrycks et al., 2022; Liu et al., 2020), the embedding space (e.g., distances to class means or nearest neighbors) (Lee et al., 2018; Ren et al., 2021; Sun et al., 2022), or a combination of both (Wang et al., 2022; Sun et al., 2021; Liu & Qin, 2023; Park et al., 2023). Much of the OSR literature focuses on how these signals are converted into an unknown score. However, both ultimately depend on the representation of the known classes. This makes representation quality particularly important when transferring pretrained embeddings to OSR.

Learning representations for open-set recognition. Rather than only changing the rule used to detect unknowns, another line of work improves the representation on which this decision is made. Contrastive, metric-learning, and prototype-based objectives encourage samples from the same known class to form compact groups while separating different known classes (Sehwag et al., 2021; Sun et al., 2022; Ming et al., 2023; Miller et al., 2021; Lu et al., 2024). Such structure can make the embedding space more informative for OSR, where unknown detection depends on whether a test sample is consistent with the known classes. These approaches typically learn or modify the representation together with the underlying model. In contrast, we ask whether an already extracted embedding can be adapted for OSR without modifying the model that produced it.

Adapting pretrained representations. A common way to transfer a pretrained representation to a downstream task is linear probing. It learns a task-specific classifier while leaving the embedding unchanged. This distinction is important for OSR: the classifier outputs become task-specific, but methods operating directly on embedding geometry still use the original representation. Recent work has therefore explored adapting the pretrained embedding space itself (Gao et al., 2024; Zhao, 2026; Zhu et al., 2024; Mannix & Bondell, 2023). However, adaptation to the known classes does not necessarily improve unknown detection. It can pull unknown samples toward known-class clusters (Zhao, 2026), while CIDER Ming et al. (2023) and PALM Lu et al. (2024) can perform worse than the original pretrained representation on near-OOD data when applied after feature extraction (Mannix & Bondell, 2023). These results motivate adaptation objectives that improve known-class structure without degrading its usefulness for unknown detection.

Adapting pretrained models. Pretrained representations can also be specialized by adapting the model that produces them. Existing approaches range from full fine-tuning (Fort et al., 2021) to prompt and parameter-efficient tuning (Miyai et al., 2023; Ming & Li, 2024), and retraining the final encoder blocks (Zhao et al., 2025). These methods can reshape the representation during downstream training, but require access to and modification of the pretrained model. Fine-tuning can also distort useful pretrained features (Kumar et al., 2022). This creates a trade-off between the flexibility of modifying the pretrained model and the constraints of adapting only its output representation.

Factors affecting open-set performance. Open-set performance is closely related to closed-set recognition. Stronger closed-set classifiers tend to perform better at detecting unknowns (Vaze et al., 2022; Galil et al., 2023), making improved classification a potential explanation for gains in OSR. The difficulty of unknown detection also depends on the relationship between known and unknown classes: rejection becomes harder as unknown classes become semantically or visually closer to the known classes (Vaze et al., 2022; Lang et al., 2024).

3 Hybrid Embedding Adaptation

Hybrid Embedding Adaptation (HEAda) is a lightweight framework for adapting existing deep embeddings to downstream open- and closed-set recognition. It operates directly on extracted embeddings, independent of the pretrained model that produced them, and learns a nonlinear transformation using both categorical and metric-learning supervision. The transformed embedding is also used to learn a classifier over the known classes, providing both an adapted representation and task-specific logits for post-hoc OSR. HEAda is not tied to a particular encoder, classification loss, or metric-learning objective.

Problem definition. Let $\mathbb{X}$ denote the input space, $\mathbb{Y}_k = \{1, \ldots, C\}$ the set of known classes available during training, and $\mathbb{Y}_u$ a disjoint set of unknown classes. Training data are drawn only from $\mathbb{X} \times \mathbb{Y}_k$, while at inference an input may belong to either $\mathbb{Y}_k$ or $\mathbb{Y}_u$. The goal is to correctly classify samples from $\mathbb{Y}_k$ while identifying samples from $\mathbb{Y}_u$ as unknown.

We study this problem in the feature-transfer setting, where each input is represented by an embedding from a pretrained encoder $f_\theta : \mathbb{X} \to \mathbb{R}^d$. Once the embeddings are extracted, the encoder is no longer accessed, and all downstream learning is performed directly in the embedding space. For an input $\boldsymbol{x} \in \mathbb{X}$, let $\boldsymbol{z} = f_\theta(\boldsymbol{x}) \in \mathbb{R}^d$ denote its extracted embedding. HEAda learns a transformation $g_\phi : \mathbb{R}^d \to \mathbb{R}^e$ and a classifier $c_\psi : \mathbb{R}^e \to \mathbb{R}^C$:

$$\boldsymbol{h} = g_\phi(\boldsymbol{z}), \qquad \boldsymbol{\ell} = c_\psi(\boldsymbol{h}), \tag{1}$$

where $\boldsymbol{h}$ is the adapted embedding and $\boldsymbol{\ell}$ contains the logits over the C known classes. Post-hoc OSR methods can then operate on $\boldsymbol{h}$, $\boldsymbol{\ell}$, or both.

Hybrid supervision. HEAda adapts the embedding space using both categorical and metric-learning supervision from the known-class training data. The categorical objective operates on the classifier outputs, while the metric-learning objective operates directly on the adapted embeddings:

$$\mathcal{L} = \lambda_{\text{cat}}\, \mathcal{L}_{\text{cat}}(\boldsymbol{\ell}, y) + \lambda_{\text{metric}}\, \mathcal{L}_{\text{metric}}(\boldsymbol{h}, y), \tag{2}$$

where λ_{cat} and λ_{metric} control their relative contributions. This formulation allows the representation to be optimized simultaneously through the classifier and directly in the embedding space, which is particularly relevant for OSR because post-hoc scoring methods may operate on either representation.

Downstream recognition. After training, HEAda maps an extracted embedding $\boldsymbol{z}$ to the adapted embedding $\boldsymbol{h}$ and corresponding known-class logits $\boldsymbol{\ell}$. For closed-set recognition, predictions are obtained from the classifier outputs $\boldsymbol{\ell}$. For open-set recognition, existing embedding-based, logit-based, and hybrid scoring methods can operate on $\boldsymbol{h}$, $\boldsymbol{\ell}$, or both. HEAda therefore adapts the representation used for downstream recognition without prescribing a particular OSR scoring rule.

Experimental setup. Unless stated otherwise, we instantiate g_ϕ as a two-layer MLP with $4\,096$ hidden and output units and train HEAda with Seesaw Loss (Wang et al., 2021) and Proxy-Anchor Loss (Kim et al., 2020) as the categorical and metric-learning objectives, respectively. We evaluate HEAda using six pretrained encoders and twelve post-hoc OSR scoring methods that operate on embeddings, logits, or both. We compare against original embeddings and linear probing and report AUROC and OSCR for OSR and accuracy for closed-set recognition. We separately study the adaptation architecture and the categorical and metric-learning objectives in Section 5.2. Full experimental details are provided in Appendix A.

4 Main Results

Much of OSR focuses on what happens at the end of the pipeline: given a representation, how should we score whether a sample is known or unknown? Our results show that there is substantial room for improvement one step earlier. Adapting the representation itself with HEAda consistently improves OSR, and the largest gains come from methods that directly use the adapted embedding.

These gains do not disappear with stronger pretrained representations, nor do they come at the cost of recognizing known classes. Even our strongest encoders benefit from HEAda, while closed-set accuracy improves at the same time, particularly for rare classes. We present the open- and closed-set results below and analyze the gains in Section 5. Full results are also available interactively.

4.1 Open-set recognition

HEAda improves OSR across encoders, datasets, and scoring methods (Figures 2 and 3). The largest gains come from methods that operate directly on the embedding. Across the 30 encoder–setting pairs, KNN gains 10.1 AUROC points on average and Mahalanobis gains 7.8 points. Grouped across embedding-based OSR methods, mean AUROC increases from 70.4 with the linear probe to 79.2 with HEAda, corresponding to a 30% reduction in the remaining AUROC error (Table 2). The improvement also carries over to FPR95, where the relative reduction is 12%.

Stronger representations still benefit from adaptation. DINO-based representations are the strongest overall, but their advantage does not make adaptation redundant. DINOv3-7B, our strongest representation, still shows substantial gains with HEAda across the open-set settings in Figure 2. The same pattern holds across the broader encoder comparison: better pretrained representations provide a stronger starting point, while HEAda further adapts them to the downstream task. Section 5.1 examines these gains with the encoder, split, and OSR method held fixed.

The scoring rule matters less once the embedding is adapted. The improvement is broad, but not uniform across OSR methods. Embedding-based methods reduce the remaining AUROC error by 30%, compared with 18% for methods using both embeddings and logits and 7% for logit-based methods (Table 2). Figure 3 shows the same pattern in the paired comparisons: KNN, Mahalanobis, and Residual improve consistently, while the logit-based methods show smaller and less consistent gains. This is notable because none of these OSR scores is part of the HEAda training objective. Appendix D further shows that the KNN and Mahalanobis gains remain positive when both settings and encoders are resampled, whereas the intervals for the evaluated logit-based methods cross zero.

The ablations help explain this difference. Section 5.2 shows that a nonlinear classification head alone does not reproduce the embedding-based OSR gains; metric supervision accounts for most of the improvement. Section 5.3 shows that the gains are largest for unknowns close to the known classes, and Appendix E finds that increased relative known–unknown separation is the geometric change most consistently associated with the KNN and Mahalanobis gains. Together, these results suggest that HEAda primarily improves OSR by changing the embedding on which the post-hoc methods operate, rather than by favoring a particular scoring rule.

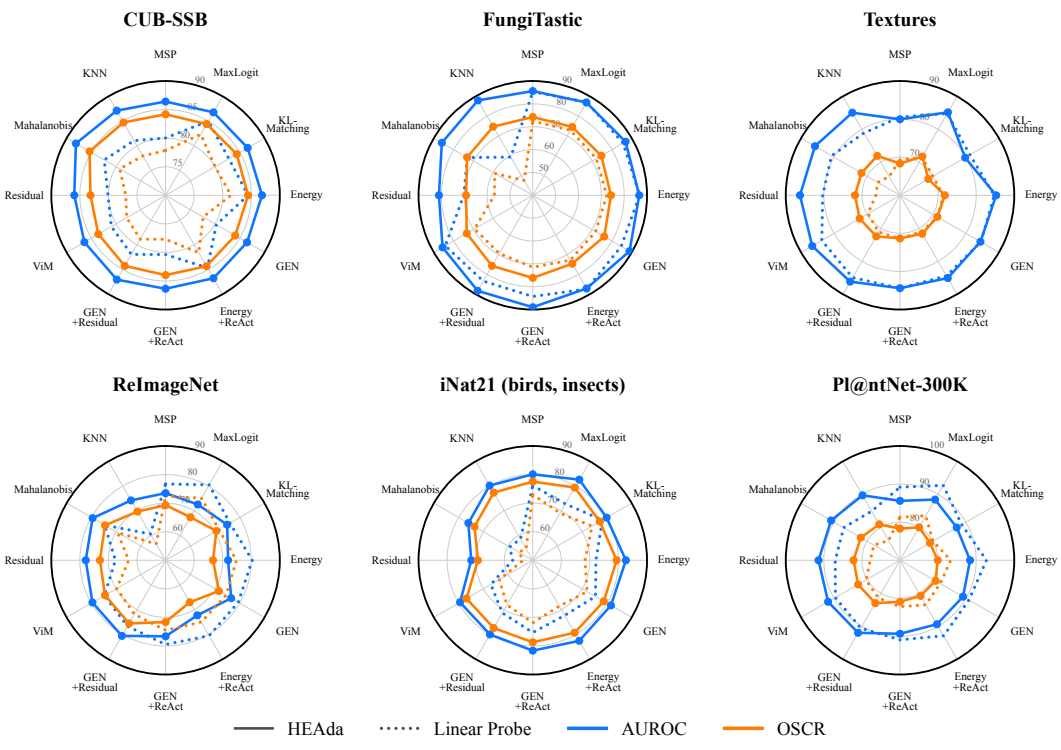


Figure 2: **Per-method OSR results of HEAda and the linear probe on DINOv3-7B.** AUROC (blue) and OSCR (orange) of the twelve post-hoc OSR methods for HEAda (solid) and the linear probe (dotted), mean of three seeds; the radial range differs per dataset. iNat21 averages the birds and insects settings of iNat2021-OSR (Lang et al., 2024) with 1-hop unknowns, and Pl@ntNet-300K (Garcin et al., 2021) uses all unknowns of its taxonomic split. Appendix A.1 describes all settings.

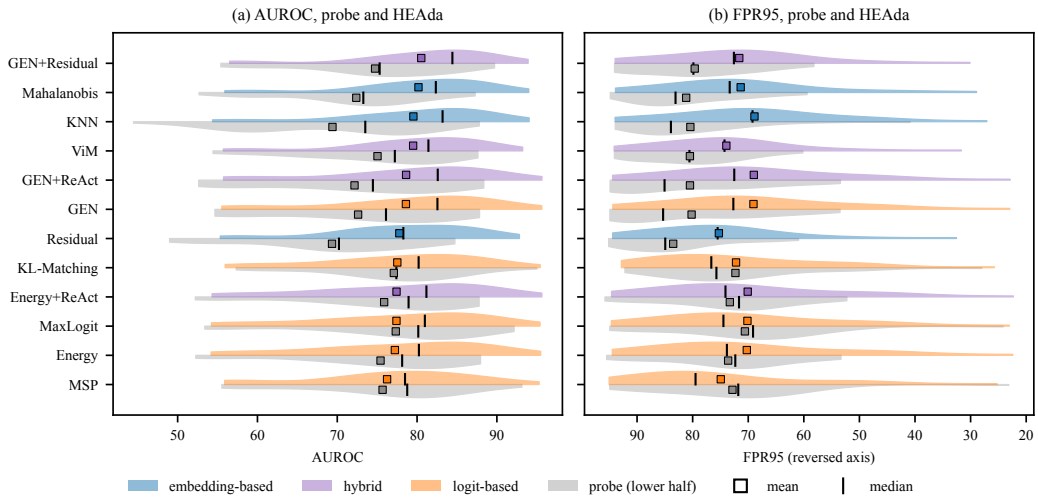


Figure 3: **Effect of HEAda on multiple post-hoc OSR methods.** Split violins compare the linear probe (gray) and HEAda (color) for 30 encoder–setting pairs, averaged over three seeds. We report (a) AUROC and (b) FPR95; higher is better in both panels, with the FPR95 axis reversed. Embedding-based OSR methods improve both ranking quality and the strict operating point, whereas logit-based OSR methods are less consistent.

4.2 Closed-set recognition

The OSR gains do not come at the expense of closed-set recognition. HEAda has higher average accuracy than the linear probe for every encoder and improves Top-1 accuracy by 3.89 points on average across the 78 encoder–dataset pairs (Table 9 in Appendix C). The improvement holds across representations with very different starting performance: averaged over the datasets of Table 9, HEAda raises BioCLIP from 63.1 to 67.9, DINOv2-giant from 80.5 to 87.5, and DINOv3-7B from 86.7 to 88.9.

A nonlinear head alone does not explain the gain. MLP+CE uses the same adaptation architecture as HEAda but is trained with cross-entropy alone. It improves over the linear probe, but remains below HEAda for every encoder; on DINOv3-7B, for example, MLP+CE reaches 87.7% compared with 88.9% for HEAda. This mirrors the OSR results: simply adding a nonlinear head helps, but does not recover the full benefit of HEAda. Section 5.2 controls for head capacity, parameter count, and the two components of the training objective to examine where the remaining gain comes from.

The largest gains are on rare classes. The difference becomes more pronounced when we look beyond Top-1 accuracy. On the three datasets with imbalanced test sets, HEAda improves balanced accuracy over the probe by 15.7 points on FungiTastic, 7.1 on FungiTastic-Mini, and 14.0 on Pl@ntNet-300K (Table 1). The corresponding macro-F1 gains are 11.3, 5.2, and 11.8 points, whereas Top-1 accuracy increases by 6.3, 1.4, and 1.0 points. The much larger gains in balanced accuracy and macro-F1 show that the closed-set improvement is concentrated on classes that contribute relatively little to overall accuracy.

Table 1: **HEAda provides its largest closed-set gains on the less common classes.** On long-tailed and fine-grained datasets, gains in mean accuracy (7.1–15.7 points) and macro-F1 (5.2–11.8) substantially exceed the Top-1 gains (1.0–6.3). Results averaged over six encoders and 3 seeds.

Dataset	Top-1			mean Accuracy			macro F1		
	LP	MLP+CE	HEAda	LP	MLP+CE	HEAda	LP	MLP+CE	HEAda
FungiTastic	66.3	70.3	**72.6**	39.0	48.8	**54.7**	32.2	39.8	**43.5**
FungiTastic-Mini	72.6	72.3	**74.0**	47.3	49.7	**54.4**	43.4	45.1	**48.6**
Pl@ntNet-300K	83.0	83.1	**84.0**	39.3	48.5	**53.3**	39.3	47.9	**51.1**

Table 2: **Every group of OSR methods improves, embedding-based OSR methods most.** Means over the family's OSR methods and the 30 encoder–setting cells (three seeds each); RER is the relative reduction of the mean error ($100 - \mathrm{AUROC}$, or FPR95) from the probe to HEAda.

		AUROC			FPR95		
OSR methods grouped by input		LP	HEAda	RER	LP	HEAda	RER
KNN, Mahalanobis, Residual	F	70.4	79.2	30 %	81.7	71.8	12 %
MSP, MaxLogit, KL-Matching, Energy, GEN	L	75.6	77.4	7 %	73.9	71.3	4 %
Energy+ReAct, GEN+ReAct/+Residual, ViM	L+F	74.5	79.0	18 %	78.5	71.2	9 %

5 Ablation Studies

The main results leave several questions about what drives the improvement. Is the gain still present when everything except the adaptation head is held fixed? Could it be explained by adding a nonlinear head, more parameters, or simply improving closed-set accuracy? Does HEAda help equally for all unknowns, or mainly when the distinction between known and unknown classes is difficult? And how sensitive are these conclusions to the encoder, OSR method, and evaluation setting?

We address these questions in turn. Section 5.1 isolates the effect of adaptation through paired comparisons, Section 5.2 dissects the source of the gain, Section 5.3 examines when adaptation helps most, and Section 5.4 tests the robustness of the result.

5.1 Q1: Does the gain hold in a controlled comparison?

The aggregate results mix different encoders, datasets, and OSR methods. We therefore ask whether the improvement remains when all three are held fixed, so that the only difference within each comparison is the adaptation head. Figure 3 compares the linear probe and HEAda over the 30 encoder–setting pairs, with each OSR method fixed in advance.

The answer is clearest for embedding-based OSR. HEAda improves KNN in 28 of 30 encoder–setting pairs and Mahalanobis in 29 of 30, with mean AUROC gains of 10.1 and 7.8 points, respectively. Residual improves in 28 of 30 pairs, with a mean gain of 8.4 points. The gains are visible both in AUROC and at the stricter FPR95 operating point (Figure 3).

The effect is weaker for methods based on classifier logits. As summarized in Table 2, embedding-based methods reduce the remaining AUROC error by 30%, compared with 18% for hybrid methods and 7% for logit-based methods. Appendix D further shows that the KNN and Mahalanobis gains remain positive when both settings and encoders are resampled, whereas the intervals for the evaluated logit-based methods cross zero.

With the encoder, setting, and method fixed, HEAda consistently improves OSR.

5.2 Q2: What drives the gain?

The controlled comparison establishes that the improvement comes from changing the adaptation head, but it does not tell us what about HEAda matters. We test four possible explanations: the nonlinear architecture, the number of trainable parameters, improved closed-set accuracy, and the metric-learning term in the objective.

Nonlinearity alone is not enough. MLP+CE uses the same width and depth as HEAda but removes the metric-learning term. It improves KNN over the linear probe on CUB-SSB ($80.5 \rightarrow 84.6$), FungiTastic ($63.7 \rightarrow 69.5$), and Textures ($76.8 \rightarrow 79.6$), but remains well below HEAda (88.2, 84.7, and 83.3, respectively), averaged over the six encoders. Higher closed-set accuracy does not explain this gap. On DINOv3-7B, MLP+CE adds 1.0 closed-set accuracy point over the probe and HEAda adds another 1.2 points, yet on FungiTastic the corresponding KNN AUROC gains are 11.3 and 17.4 points. Even after ℓ_2-normalizing the MLP+CE embedding as in HEAda, it closes only part of the Mahalanobis gap and none of the KNN gap. Thus, the OSR improvement cannot be reproduced simply by adding a nonlinear classification head.

More parameters do not explain the gain. We next narrow the adapter until the complete head has the same number of parameters, $DC + C$, as the linear probe. Across six encoders and four settings, this parameter-matched HEAda still outperforms the probe in $20/24$ encoder–setting pairs with KNN and $22/24$ with Mahalanobis, with median gains of 6.7 and 6.8 AUROC points. It also remains close to the full HEAda, trailing it by a median of only 1.9 points with KNN and 0.2 with Mahalanobis. The main exception is BioCLIP, where matching the probe parameter count reduces the adapted embedding to only 31–226 dimensions. On DINOv3-7B FungiTastic, where the matched head has 11.6M parameters, KNN still improves from 59.3 to 86.2, compared with 88.0 for the full HEAda. These results rule out parameter count alone as an explanation for the embedding-based gains.

Metric supervision accounts for most of the OSR gain. Finally, we separate the two components of the HEAda objective. Only heads trained with Proxy-Anchor improve the best achievable AUROC over the linear probe (Table 3); classification-only heads do not. Proxy-Anchor alone and full HEAda achieve similar best AUROC, while combining metric and categorical supervision gives the strongest OSCR. Appendix B extends this analysis across losses and shows that the gain is not specific to HEAda's categorical loss.

The improvement is not explained by nonlinearity, parameter count, or higher closed-set accuracy; metric supervision accounts for most of the OSR gain.

Table 3: **Metric supervision drives the OSR gain.** Results on DINOv3-7B, averaged over five open-set settings and three seeds. For each head, we report the best AUROC and OSCR across the twelve post-hoc OSR methods. †The Proxy-Anchor-only head classifies using its nearest proxy.

training objective	best AUROC	Δ vs probe	best OSCR
LP + CE	83.37	0	73.49
LP + frequency margin	83.12	−0.25	73.33
MLP + CE	82.51	−0.86	72.21
MLP + frequency margin	83.10	−0.27	73.44
MLP + Proxy-Anchor†	**84.78**	+1.41	74.55
MLP + both (HEAda)	84.63	+1.26	**75.28**

5.3 Q3: When does adaptation help most?

The gains from HEAda vary across open-set settings. We find a consistent pattern: adaptation helps most when unknowns are close to the known classes. We examine this in two complementary ways, using taxonomic distance, where proximity is defined independently of the representation, and embedding geometry, where we measure how well known and unknown samples are separated.

The gains are largest for taxonomically close unknowns. iNat2021-OSR (Lang et al., 2024) provides a controlled way to vary the semantic difficulty of the unknown set. At hop 1, an unknown species shares its genus with a known species, whereas at hop 7 the unknown comes from a different kingdom. Across all six encoders, the KNN gain from HEAda is largest at hops 1–3, reaching 4.3–26.2 AUROC points at hop 1, and becomes small at hops 5–7, where the probe already separates most unknowns and the two heads differ by less than one AUROC point. The same trend appears in the SSB difficulty tiers: the Mahalanobis gain increases from 3.2 AUROC points for Easy unknowns to 10.4 for Hard unknowns, while the KNN gain increases from 0.6 to 10.4 (Table 4). The benefit of adaptation is therefore largest where the original representation has difficulty separating unknowns from the known classes.

Table 4: **Pre-specified OSR method gains grow from Easy to Hard unknowns for KNN and Mahalanobis, while KL-Matching loses on Easy.** Values are HEAda minus probe AUROC for seed 3407, averaged over both SSB settings and six encoders with each OSR method held fixed.

Difficulty	Mahalanobis	KNN	GEN+Residual	KL-Matching	MaxLogit
Easy Δ	+3.19	+0.62	+6.56	−3.85	+0.89
Medium Δ	+8.16	+7.19	+8.57	−0.50	+3.00
Hard Δ	+10.44	+10.38	+7.86	+4.97	+2.98

Relative known–unknown separation tracks the gain. Taxonomic proximity gives us an external notion of difficulty, but we can also ask whether the same pattern is visible directly in the embedding space. HEAda substantially changes both known-class structure and the position of unknown samples relative to it. Across encoder–setting–seed cells, the geometric statistic most consistently associated with the KNN and Mahalanobis gains is the separation ratio: the distance of an unknown sample to its nearest known class relative to the within-class distance of known samples. In contrast, changes in known-class compactness alone do not show a consistent relationship with the OSR gain. Appendix E provides the full analysis, including correlations across seeds and leave-one-setting-out checks. These two views tell a consistent story. HEAda provides the largest gains when the unknowns are difficult to separate from the known classes, and its improvement is associated more strongly with increased relative known–unknown separation than with simply making the known classes more compact.

HEAda helps most where OSR is hardest: when unknowns lie close to the known classes.

5.4 Q4: How robust are the gains?

The previous analyses use multiple encoders and OSR methods, but their measurements are not independent: the same open-set settings are evaluated across several encoders. We therefore test whether the main conclusions remain when these sources of variation are accounted for.

The gains persist across encoders and settings. HEAda improves embedding-based OSR across all evaluated encoders, although the magnitude of the improvement varies. To account for the shared evaluation settings, we bootstrap over both settings and encoders. The resulting AUROC gains remain positive for KNN ($+10.1$; 95% CI $[4.7, 18.0]$) and Mahalanobis ($+7.8$; $[4.5, 11.7]$). In contrast, the intervals for the evaluated logit-based methods cross zero. Appendix D provides the full analysis.

The result does not require selecting an OSR method per setting. We also ask whether HEAda's performance depends on choosing the best OSR method for each target setting. In a leave-one-setting-out analysis, fixing Mahalanobis in advance comes within 1.10 AUROC points of the per-setting oracle, while fixed KNN comes within 1.74 points. Selecting an OSR method separately for each encoder does not improve this gap. Thus, the strong embedding-based results do not depend on tuning the scoring method to each test setting.

The choice of unknown classes matters more than the training seed. On Textures, where unknown classes are selected by class holdout, we repeat the experiment with five fixed known–unknown splits and three training seeds per split. Variation across class splits is 2.0–$2.3\times$ larger than variation across seeds for HEAda, and 17–$26\times$ larger for the linear probe. The composition of the unknown set is therefore a more important source of variation than repeated optimization (Appendix D).

Overall, the embedding-based gains remain consistent across encoders and evaluation settings and do not depend on selecting an OSR method for each target setting.

6 Conclusion

We asked how much task-specific adaptation strong pretrained representations still need for open-set recognition. Adapting the extracted embeddings with HEAda, a lightweight head trained on known classes with a hybrid classification and metric-learning objective, leaves the encoder untouched and still improves every group of post-hoc OSR methods, reducing the mean AUROC error by $30\,\%$ for embedding-based, $18\,\%$ for hybrid, and $7\,\%$ for logit-based methods. After adaptation, simple KNN and Mahalanobis scores become among the strongest OSR methods, so improving the representation can matter more than refining the scoring rule. The gain comes mainly from the metric term rather than from extra capacity, is largest for unknowns that are semantically close to the known classes, and carries over to closed-set recognition, where it is most pronounced on rare classes. Logit-based methods remain sensitive to how a head is trained, as ReImageNet shows, so a post-hoc OSR method should be evaluated on the representation it will read.

AI USE STATEMENT

We used generative AI tools, accessed through a coding assistant, to edit and reformulate text, generate LaTeX tables from experiment logs, and assist with plotting scripts over results that had already been computed. The tools were not used to produce or select experimental measurements or to write the training and evaluation code that generated the reported model outputs. Reported values are derived from the experiment logs and result files. We take responsibility for the study design, verification, interpretation, and final content, including AI-assisted text and artifacts.

ETHICS STATEMENT

The work uses established research image datasets (CUB, FGVC-Aircraft, DTD, FungiTastic, iNaturalist 2021, Pl@ntNet-300K, ImageNet-1k, and the ReImageNet re-annotation) and released pretrained encoders under their respective access terms and licenses; no new data were collected and no human subjects were involved. The method inherits the potential biases and misuse risks of its datasets, pretrained encoders, and image-classification applications.

REPRODUCIBILITY STATEMENT

To allow full reproducibility, we provide the implementation and experimental details needed to reproduce all results reported in this paper. Section 3 describes the HEADA architecture, training objectives, hyperparameters, and checkpoint selection, while Appendix A describes the pretrained encoders, data splits, and evaluation protocol. All heads are trained on precomputed embeddings, so the pretrained encoders do not need to be rerun during training. The appendix provides the complete settings for the OSR scoring methods, additional experimental details, and seed variability. The data splits, trained heads, result tables, and evaluation and analysis code are available in an anonymous repository: `https://anonymous.4open.science/r/HEAD/`.

REFERENCES

Nikolas Adaloglou, Felix Michels, Tim Kaiser, and Markus Kollmann. Adapting contrastive language-image pretrained (clip) models for out-of-distribution detection. *arXiv preprint arXiv:2303.05828*, 2023.

Akshay Raj Dhamija, Manuel Günther, and Terrance Boult. Reducing network agnostophobia. *Advances in Neural Information Processing Systems*, 31, 2018.

Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. Exploring the limits of out-of-distribution detection. In *Advances in Neural Information Processing Systems*, volume 34, pp. 7068–7081, 2021.

Ido Galil, Mohammed Dabbah, and Ran El-Yaniv. A framework for benchmarking class-out-of-distribution detection and its application to imagenet. In *International Conference on Learning Representations (ICLR)*, 2023.

Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *International journal of computer vision*, 132(2):581–595, 2024.

Camille Garcin, alexis joly, Pierre Bonnet, Antoine Affouard, Jean-Christophe Lombardo, Mathias Chouet, Maximilien Servajean, Titouan Lorieul, and Joseph Salmon. Pl@ntnet-300k: a plant image dataset with high label ambiguity and a long-tailed distribution. In J. Vanschoren and S. Yeung (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021.

Jianyang Gu, Samuel Stevens, Elizabeth Campolongo, Matthew Thompson, Net Zhang, Jiaman Wu, Andrei Kopanev, Zheda Mai, Alexander White, James Balhoff, et al. Bioclip 2: Emergent properties from scaling hierarchical contrastive learning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 102778–102811, 2025.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations (ICLR)*, 2017.

Dan Hendrycks, Steven Basart, Mantas Mazeika, Andy Zou, Joe Kwon, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *PMLR*, 2022.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2790–2799. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/houlsby19a.html.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (eds.), *Computer Vision – ECCV 2022*, pp. 709–727, Cham, 2022. Springer Nature Switzerland. ISBN 978-3-031-19827-4.

Sungyeon Kim, Dongwon Kim, Minsu Cho, and Suha Kwak. Proxy anchor loss for deep metric learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3235–3244, 2020. doi: 10.1109/CVPR42600.2020.00330.

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations (ICLR)*, 2022.

Nico Lang, Vésteinn Snæbjarnarson, Elijah Cole, Oisin Mac Aodha, Christian Igel, and Serge Belongie. From coarse to fine-grained open-set recognition. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17804–17814, 2024. doi: 10.1109/CVPR52733.2024.01686.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/abdeb6f575ac5c6676b747bca8d09cc2-Paper.pdf.

Xiwen Li, Wei Quan, Jia Huang, and Peiyuan Hong. Recognizing unknowns: a survey on visual open-set recognition. *Applied Intelligence*, 55(16):1055, October 2025. ISSN 1573-7497. doi: 10.1007/s10489-025-06956-7.

Litian Liu and Yao Qin. Fast decision boundary based out-of-distribution detector. *arXiv preprint arXiv:2312.11536*, 2023.

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 21464–21475. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/f5496252609c43eb8a3d147ab9b9c006-Paper.pdf.

Xixi Liu, Yaroslava Lochman, and Christopher Zach. Gen: Pushing the limits of softmax-based out-of-distribution detection. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23946–23955, 2023. doi: 10.1109/CVPR52729.2023.02293.

Haodong Lu, Dong Gong, Shuo Wang, Jason Xue, Lina Yao, and Kristen Moore. Learning with mixture of prototypes for out-of-distribution detection. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 8613–8635, 2024.

Evelyn Mannix and Howard Bondell. A mixture of exemplars approach for efficient out-of-distribution detection with foundation models. *arXiv preprint arXiv:2311.17093*, 2023.

Thomas Mensink, Jakob Verbeek, Florent Perronnin, and Gabriela Csurka. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2624–2637, 2013.

Dimity Miller, Niko Sunderhauf, Michael Milford, and Feras Dayoub. Class anchor clustering: A loss for distance-based open set recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3570–3578, 2021.

Yifei Ming and Yixuan Li. How does fine-tuning impact out-of-distribution detection for vision-language models? *International Journal of Computer Vision*, 132(2):596–609, 2024.

Yifei Ming, Yiyou Sun, Ousmane Dia, and Yixuan Li. How to exploit hyperspherical embeddings for out-of-distribution detection? In *International Conference on Learning Representations (ICLR)*, 2023.

Atsuyuki Miyai, Qing Yu, Go Irie, and Kiyoharu Aizawa. Locoop: Few-shot out-of-distribution detection via prompt learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 76298–76310, 2023.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

Jaewoo Park, Yoon Gyo Jung, and Andrew Beng Jin Teoh. Nearest neighbor guidance for out-of-distribution detection. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1686–1695. IEEE, 2023.

Lukas Picek, Klara Janouskova, Vojtech Cermak, and Jiri Matas. Fungitastic: A multi-modal dataset and benchmark for image categorization. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2037–2047. IEEE, 2025. doi: 10.1109/CVPRW67362.2025.00192.

Jie Ren, Stanislav Fort, Jeremiah Liu, Abhijit Guha Roy, Shreyas Padhy, and Balaji Lakshminarayanan. A simple fix to mahalanobis distance for improving near-ood detection. *arXiv preprint arXiv:2106.09022*, 2021.

Ruslan Rozumnyi, Matěj Suchánek, Tomáš Vojíř, Klára Janoušková, and Jiří Matas. Fi-imagenet-1k: An ood benchmark from the inside of the imagenet-1k validation set. *arXiv preprint arXiv:2609.01027*, 2026.

Walter J Scheirer, Anderson de Rezende Rocha, Archana Sapkota, and Terrance E Boult. Toward open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(7): 1757–1772, 2013.

Vikash Sehwag, Mung Chiang, and Prateek Mittal. Ssd: A unified framework for self-supervised outlier detection. In *International Conference on Learning Representations (ICLR)*, 2021.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. Dinov3, 2025.

Samuel Stevens, Jiaman Wu, Matthew J Thompson, Elizabeth G Campolongo, Chan Hee Song, David Edward Carlyn, Li Dong, Wasila M Dahdul, Charles Stewart, Tanya Berger-Wolf, et al. Bioclip: A vision foundation model for the tree of life. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19412–19424. IEEE, 2024.

Yiyou Sun, Chuan Guo, and Yixuan Li. React: Out-of-distribution detection with rectified activations. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 144–157. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/01894d6f048493d2cacde3c579c315a3-Paper.pdf.

Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 20827–20840. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/sun22d.html.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features, 2025.

Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Open-set recognition: A good closed-set classifier is all you need. In *International conference on learning representations*, 2022.

Illia Volkov, Nikita Kisel, Tetiana Mishkina, Klara Janouskova, and Jiri Matas. Doomed to re-annotate, forever: The imagenet story. *arXiv preprint arXiv:2608.13783*, 2026.

Haoqi Wang, Zhizhong Li, Litong Feng, and Wayne Zhang. Vim: Out-of-distribution with virtual-logit matching. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4911–4920. IEEE, 2022.

Jiaqi Wang, Wenwei Zhang, Yuhang Zang, Yuhang Cao, Jiangmiao Pang, Tao Gong, Kai Chen, Ziwei Liu, Chen Change Loy, and Dahua Lin. Seesaw loss for long-tailed instance segmentation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9690–9699. IEEE, 2021.

Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *International Journal of Computer Vision*, 132(12):5635–5662, June 2024. ISSN 1573-1405. doi: 10.1007/s11263-024-02117-4.

Jingyang Zhang, Jingkang Yang, Pengyun Wang, Haoqi Wang, Yueqian Lin, Haoran Zhang, Yiyou Sun, Xuefeng Du, Yixuan Li, Ziwei Liu, et al. Openood v1. 5: Enhanced benchmark for out-of-distribution detection. *arXiv preprint arXiv:2306.09301*, 2023.

Dongfang Zhao. Ega: Adapting frozen encoders for vector search with bounded out-of-distribution degradation. *arXiv preprint arXiv:2605.05674*, 2026.

Shizhen Zhao, Jiahui Liu, Xin Wen, Haoru Tan, and Xiaojuan Qi. Equipping vision foundation model with mixture of experts for out-of-distribution detection. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1751–1761. IEEE, 2025.

Lin Zhu, Yifeng Yang, Qinying Gu, Xinbing Wang, Chenghu Zhou, and Nanyang Ye. Croft: robust fine-tuning with concurrent optimization for ood generalization and open-set ood detection. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24, 2024.

A Experimental Details

Our experiments follow the same feature-transfer setup throughout. We first extract and freeze features from a pretrained vision encoder, then train either a linear classifier or an adaptation head using known-class data only. Open-set detection is performed afterward using standard post-hoc scoring methods, without updating the encoder or using unknown samples during training. Below, we describe the datasets and open-set splits, pretrained encoders, adaptation and baseline configurations, OSR methods, and evaluation protocol.

A.1 Datasets and Open-Set Protocols

Open-set evaluation depends heavily on how unknown samples are defined. We therefore evaluate HEAda across protocols that introduce unknowns in different ways, including temporal, semantic, taxonomic, and random class holdouts, as well as naturally out-of-label-space images. This gives us settings that range from fine-grained unknowns that are closely related to the known classes to broader distribution shifts. Table 5 summarizes the resulting open-set settings and their sizes. For closed-set evaluation, we follow the standard splits.

FungiTastic. We follow the temporal open-set protocol of FungiTastic (Picek et al., 2025). Training is restricted to observations collected through 2021, followed by validation data from 2022 and test data from 2023. Species absent from the training are treated as unknown at test time. Unlike a random class holdout, unknown species therefore come naturally as new observations are collected.

CUB-SSB. We follow the known/unknown class partition from the Semantic Shift Benchmark (SSB) (Vaze et al., 2022). We construct the image-level splits for the known classes and use all images from the designated unknown classes for open-set evaluation. This gives us a fine-grained OSR setting in which both known and unknown classes are bird species.

iNat2021-OSR. iNat2021-OSR (Lang et al., 2024) defines taxonomic open-set splits for birds and insects, with unknown species selected at different taxonomic distances from the known classes. Our main evaluation uses hop 1, the most fine-grained setting, where every unknown species shares a genus with at least one known species.

Textures. Textures does not provide a predefined open-set protocol. We therefore hold out a random subset of 10 of its 47 classes as unknown. We generate this partition once and keep it fixed across all encoders, methods, and random seeds.

ReImageNet. ReImageNet (Volkov et al., 2026) provides a different type of open-set evaluation in which unknowns are individual images rather than held-out classes from the same dataset. The unknown set contains 655 images identified by Fi-ImageNet-1k as lying outside the ImageNet-1k label space (Rozumnyi et al., 2026). The known set is formed from a stratified half of the images.

Pl@ntNet-300K. Pl@ntNet-300K (Garcin et al., 2021) is a long-tailed plant-recognition dataset for which we construct a taxonomic open-set split. We hold out complete species at different taxonomic distances from the known classes, ranging from sister species within a known genus to species from different orders. We generate this partition once and keep it fixed across all experiments. Unless otherwise stated, we report results aggregated across all unknown taxonomic distances.

Table 5: **Open-set evaluation settings and dataset sizes.** We report the number of known classes, train, validation, and test examples, and the number of unknown examples in the test set.

Setting	known classes	train	validation	total test	unknown in test	split
FungiTastic open-set	2,829	433,702	89,659	97,551	5,719	temporal (Picek et al., 2025)
CUB-SSB	100	2,004	1,556	8,228	5,907	SSB classes (Vaze et al., 2022)
iNat2021-OSR birds	745	191,698	18,625	10,420	2,970	taxonomic (Lang et al., 2024)
iNat2021-OSR insects	1,501	361,427	37,525	20,060	5,050	taxonomic (Lang et al., 2024)
Textures	37	1,508	1,508	2,624	1,116	random holdout
ReImageNet	1,000	1,131,167	150,000	24,882	655	Fi-ImageNet-1k (Rozumnyi et al., 2026)
Pl@ntNet-300K	462	98,104	12,535	31,112	18,583	taxonomic (ours)

A.2 Evaluation Protocol

We evaluate each trained head under both closed-set and open-set recognition. Closed-set evaluation measures classification performance on the known classes, while open-set evaluation additionally tests whether known and unknown samples can be separated. For open-set evaluation, known samples come from classes available during training, while unknown samples follow the dataset-specific protocols described in Section A.1. The same data splits are used for linear probe and HEAda.

Each model is trained for 50 epochs and evaluated at the final epoch. We repeat every experiment with three random seeds: $3{,}407$, $1{,}234$, and 42, and report the mean across seeds.

In our setting, unknown samples are never used to train the head, select its hyperparameters, or calibrate the post-hoc OSR scores. Both the linear probe and HEAda are therefore trained and selected using only known-class data and evaluated on the same open-set splits. This ensures that HEAda's OSR gains cannot be attributed to access to the unknown test distribution.

Metrics. We evaluate closed-set recognition using Top-1 accuracy and open-set recognition using AUROC, OSCR (Dhamija et al., 2018), and FPR95. AUROC measures how well the OSR score separates known from unknown samples across thresholds. OSCR additionally accounts for whether known samples are classified correctly, connecting unknown rejection with closed-set recognition. FPR95 measures the false-positive rate when 95% of known samples are retained. Higher values are better for Top-1 accuracy, AUROC, and OSCR, while lower values are better for FPR95.

Aggregation. We first average results across the three random seeds for each encoder–dataset–method combination. For results aggregated across encoders and datasets, we then give each encoder–dataset combination equal weight. Because these combinations reuse the same datasets and encoders, we treat the resulting averages as descriptive summaries rather than independent experiments.

Relative error reduction. For AUROC, we additionally report relative error reduction (RER), where the remaining error is defined as $1 - \mathrm{AUROC}$:

$$\mathrm{RER} = \frac{\mathrm{AUROC}_{\mathrm{HEAda}} - \mathrm{AUROC}_{\mathrm{LP}}}{1 - \mathrm{AUROC}_{\mathrm{LP}}}. \tag{3}$$

For grouped results, we first average AUROC across the OSR methods in the group and across encoder–dataset combinations following the aggregation procedure above, and then compute RER from the resulting LP and HEAda means. We do not compute RER independently for each cell and then average the resulting percentages. Relative FPR95 reductions are computed in the same way from the corresponding aggregated FPR95 values.

A.3 Feature Adaptation and Baselines

We compare three ways of using the frozen encoder features: (i) the original embeddings without task-specific adaptation (i.e., a linear probe), (ii) an MLP trained with categorical and metric supervision and their combination, and (iii) HEAda. All three use the same encoder features.

Original embeddings. We evaluate the extracted encoder embeddings directly, without learning a task-specific transformation. Embedding-based OSR methods operate directly on these features. Because logit-based and hybrid OSR methods require classifier outputs, they use the logits from the linear probe described below while retaining the original embeddings.

Linear probe. Our primary baseline trains a linear classifier on the frozen encoder embeddings without modifying the representation. The probe is trained with cross-entropy using the same optimizer, learning-rate schedule, number of epochs, and random seeds as HEAda.

Classification-only MLP. To separate the effect of nonlinear feature adaptation from the metric-learning objective, we train the same adaptation head using classification supervision only. This baseline has the same size and training setup as HEAda but omits $\mathcal{L}_{\mathrm{metric}}$.

HEAda architecture. For the main experiments, g_ϕ consists of a linear layer with $4{,}096$ output units, followed by LayerNorm, GELU, dropout with rate 0.5, and a second linear layer. The resulting $4{,}096$-dimensional representation is ℓ_2-normalized before being passed to the linear classifier c_ψ. Unless stated otherwise, we use the same architecture for all six pretrained encoders.

HEAda objective. Our main configuration combines Seesaw Loss (Wang et al., 2021) with $p = 0.8$ for $\mathcal{L}_{\text{cat}}$ and Proxy-Anchor Loss (Kim et al., 2020) with one learned proxy per class, $\alpha = 32$, and $\delta = 0.1$ for $\mathcal{L}_{\text{metric}}$. We weight the two terms equally, $\lambda_{\text{cat}} = \lambda_{\text{metric}} = 0.5$.

Training. We train both the linear probe and HEAda for 50 epochs using AdamW with a learning rate of 10^{-3}, weight decay $5 \cdot 10^{-4}$, and batch size 256. The learning rate is warmed up linearly from 0.1 times its base value over the first five epochs and then reduced by a factor of 0.9 whenever the validation loss does not improve for one epoch. We repeat training with seeds $3{,}407$, $1{,}234$, and 42 and evaluate the final epoch. For paired comparisons between the linear probe and HEAda, we fix the encoder, dataset split, random seed, and post-hoc OSR method. The comparison therefore changes the adaptation procedure while keeping the underlying pretrained features and downstream OSR scoring rule fixed.

A.4 Post-hoc OSR Methods

We apply the same set of post-hoc OSR methods to the outputs of the linear probe and HEAda. The methods cover scores based on classifier logits, adapted features, or both. For embedding-based methods, all class-conditional statistics and reference features are estimated using known-class training data. No unknown samples are used to fit or calibrate the scoring rules. Table 6 summarizes the input and configuration of each method. We keep these configurations fixed across encoders, datasets, seeds, and head types. This lets us test whether feature adaptation improves OSR without changing the downstream scoring method.

Table 6: **Post-hoc OSR methods and their configurations.**

Method	Input	Configuration
MSP (Hendrycks & Gimpel, 2017)	L	standard
MaxLogit (Hendrycks et al., 2022)	L	standard
KL-Matching (Hendrycks et al., 2022)	L	class posteriors estimated on training data
Energy (Liu et al., 2020)	L	standard
GEN (Liu et al., 2023)	L	$\gamma = 0.1$, $M = 20$
Mahalanobis (Lee et al., 2018)	F	shared covariance pseudoinverse; no preprocessing
KNN (Sun et al., 2022)	F	$K = 1$; ℓ_2-normalized F
Residual (Wang et al., 2022)	F	512 residual dimensions ($d/2$ if $d < 768$)
ViM (Wang et al., 2022)	F + L	same residual dimensions as Residual
Energy+ReAct (Sun et al., 2021)	F + L	per-coordinate 0.95 training quantile
GEN+ReAct (Liu et al., 2023; Sun et al., 2021)	F + L	GEN with 0.95 training-quantile clipping
GEN+Residual (Liu et al., 2023)	F + L	$1{,}000$ residual dims. if $d \geq 2{,}048$, otherwise 512

B Dissecting the HEAda Objective

The main results show that HEAda improves OSR, but they leave open what drives the improvement. In particular, it is unclear whether the gain comes simply from adding a nonlinear adaptation head or from the metric supervision used to train it. We disentangle these effects by comparing classification-only, metric-only, and hybrid heads, and then examine how sensitive the result is to the particular classification and metric losses.

Metric supervision drives the OSR gain. Simply adding a nonlinear head is not enough to explain the improvement from HEAda. An MLP trained with cross-entropy alone improves KNN AUROC from 71.0 to 74.6, but decreases both Mahalanobis AUROC and OSCR. In contrast, adding metric supervision substantially improves both KNN and Mahalanobis. The Proxy-Anchor-only head and HEAda perform similarly for OSR, with their AUROC values within 0.3 points of each other. Together, these results suggest that metric supervision, rather than the nonlinear head itself, accounts for most of the OSR improvement (Table 7).

Table 7: **Where does the gain from HEAda come from?** The improvement in OSR comes from metric supervision, not the nonlinear head itself, while combining metric and categorical supervision recovers the strongest closed-set accuracy. Results averaged over 3 seeds and FungiTastic, CUB-SSB, Textures, and ReImageNet datasets. †Classifies by its nearest proxy. Balanced accuracy weights every known class equally and therefore reflects rare classes.

Training objective	kNN AUROC	Mahalanobis AUROC	kNN OSCR	Mahalanobis OSCR	Top1 Accuracy	Mean Accuracy
LL + CE	71.0	77.6	62.0	67.6	85.3	80.9
LL + frequency margin	71.0	77.6	61.9	67.6	85.3	81.5
MLP + CE	74.6	76.0	65.0	65.5	84.3	80.4
MLP + frequency margin	79.0	80.2	69.8	70.6	84.7	81.4
MLP + Proxy-Anchor†	81.9	**83.9**	72.6	73.9	84.8	79.7
MLP + both	**82.0**	83.6	**73.2**	**74.3**	**85.6**	**82.3**

Classification supervision matters for closed-set recognition. If metric supervision accounts for most of the OSR improvement, why include classification supervision at all? The difference becomes clear in closed-set recognition. Across six encoders and eleven datasets, HEAda reaches $83.2\,\%$ mean closed-set accuracy, compared with $79.7\,\%$ for the Proxy-Anchor-only head and $78.9\,\%$ for the linear probe. The gap is particularly large on the long-tailed FungiTastic dataset, where HEAda reaches $63.7\,\%$ balanced accuracy, compared with $57.4\,\%$ for the probe and $51.6\,\%$ for Proxy-Anchor alone. Thus, while metric supervision accounts for most of the OSR improvement, combining it with classification supervision gives substantially stronger closed-set recognition.

How sensitive is HEAda to the choice of losses? The previous results identify metric supervision as the main contributor to the OSR improvement, but HEAda still combines a particular metric loss with a particular classification loss. We therefore evaluate three metric-learning objectives together with five classification losses on DINOv3-7B. The choice of classification loss has little effect on the best achievable AUROC when metric supervision is fixed. Cross-entropy, Seesaw, and Focal differ by at most 0.2 AUROC points within each metric-learning objective. In particular, replacing Seesaw with standard cross-entropy leaves the performance of Proxy-Anchor nearly unchanged. The choice of metric objective has a larger effect. Proxy-Anchor achieves the highest best AUROC and OSCR with each classification loss, while Center loss reaches similar KNN performance. Overall, the ablation supports the same conclusion as the head comparison: the OSR improvement is primarily associated with metric supervision and is not specific to the Seesaw loss used in HEAda (Table 8).

Table 8: **Hybrid objective ablation.** HEAda does not rely on a particular categorical loss. In contrast, the choice of metric-learning objective has a more consistent effect on open-set performance, supporting metric supervision as the key component of the hybrid objective. *Results use DINOv3-7B and are averaged over CUB-SSB, Textures, and Pl@ntNet-300K across three seeds. Best and mean AUROC are computed over the twelve post-hoc OSR methods.*

Metric-learning loss	Categorical loss	best AUROC	mean AUROC	kNN AUROC	best OSCR	Top1 accuracy
Proxy-Anchor	CE	**89.1**	**87.4**	87.2	**80.4**	87.9
	Seesaw	88.9	86.9	**87.3**	80.3	**88.0**
	Focal	89.0	**87.4**	87.1	80.2	87.8
	LogitNorm	88.5	77.2	86.9	79.9	87.7
	logit adj.	88.7	85.6	87.2	78.0	85.6
ArcFace	CE	88.0	85.0	85.6	79.7	87.7
	Seesaw	87.9	84.6	85.6	79.6	87.6
	Focal	87.8	84.9	85.4	79.5	87.6
	LogitNorm	86.8	77.6	85.7	78.6	87.5
	logit adj.	88.0	83.8	85.7	76.7	84.5
Center	CE	87.4	84.1	**87.3**	78.9	87.5
	Seesaw	87.5	84.0	**87.3**	79.0	87.5
	Focal	87.3	83.3	87.1	78.7	87.3
	LogitNorm	86.5	74.9	86.5	78.0	87.8
	logit adj.	87.7	82.7	87.2	76.8	85.1

C Full Closed-Set Results

Closed-set accuracy of the linear probe, MLP+CE, and HEAda for selected encoders and datasets.

Table 9: **HEAda has the highest closed-set accuracy on every encoder.** Top-1 accuracy (%) of the linear probe (LP), MLP+CE, and HEAda on the full class sets and official test splits, mean of three seeds; bold marks the best head per encoder and dataset, and Avg. is over all thirteen datasets.

Encoder	Head	CUB	Aircraft	Textures	FT-Mini	FT-full	Flowers	Food	Pets	Cars	iNat	ImageNet	iNat21	Pl@ntNet	**Avg.**
DINOv3-7B	LP	92.3	69.2	**78.6**	79.3	76.4	99.6	96.6	96.9	94.1	80.5	88.2	90.0	**84.5**	86.7
	MLP+CE	92.2	81.2	77.5	79.0	78.0	99.6	96.3	97.2	**94.7**	82.5	87.2	91.4	83.8	87.7
	HEAda	**92.6**	**85.1**	77.2	**80.9**	**80.3**	**99.7**	**97.0**	**97.3**	**94.7**	**86.3**	**88.3**	**92.5**	**84.5**	**88.9**
DINOv2-G	LP	90.4	43.1	75.1	74.4	66.2	**99.7**	93.4	95.8	86.0	65.3	86.2	86.7	83.6	80.5
	MLP+CE	91.8	83.5	**75.7**	75.7	74.2	**99.7**	94.5	96.5	92.3	78.0	85.8	89.4	83.6	86.2
	HEAda	**92.5**	**86.0**	75.3	**78.1**	**76.0**	**99.7**	**95.5**	**96.7**	**93.9**	**82.4**	**86.6**	**90.3**	**84.4**	**87.5**
SigLIP2	LP	88.8	78.9	76.8	65.4	56.0	99.5	97.2	94.9	**96.5**	61.1	88.5	81.8	81.0	82.0
	MLP+CE	89.2	84.8	**79.3**	65.2	63.5	99.6	96.8	**96.8**	96.2	75.3	87.5	84.7	82.6	84.7
	HEAda	**90.5**	**85.7**	79.2	**67.3**	**66.3**	**99.7**	**97.3**	96.7	96.3	**80.4**	**88.6**	**86.2**	**83.5**	**86.0**
RADIOv2-G	LP	90.6	74.8	78.5	71.8	62.6	99.6	96.1	96.2	95.6	70.5	88.0	85.2	83.1	84.0
	MLP+CE	90.5	83.4	78.5	71.8	69.3	**99.7**	95.9	96.7	95.3	76.7	87.3	86.9	83.3	85.8
	HEAda	**91.5**	**85.9**	**78.8**	**73.2**	**71.5**	**99.7**	**96.7**	**96.9**	**95.8**	**82.5**	**88.4**	**88.8**	**84.2**	**87.2**
BioCLIP	LP	**90.1**	18.5	52.1	**65.5**	59.7	96.6	56.8	73.5	16.4	70.6	57.3	81.2	81.8	63.1
	MLP+CE	89.3	25.8	58.4	64.7	60.3	98.2	66.5	80.2	24.5	73.3	59.9	80.8	81.4	66.4
	HEAda	89.9	**27.4**	**58.9**	**65.5**	**62.9**	**98.4**	**68.5**	**80.9**	**27.8**	**78.2**	**60.7**	**81.3**	**82.9**	**67.9**
BioCLIP 2	LP	**94.0**	41.9	69.7	**79.0**	77.1	99.1	87.7	89.4	91.0	88.7	79.4	**94.0**	83.9	82.7
	MLP+CE	93.2	52.5	**74.2**	77.2	76.8	99.3	89.2	90.9	91.6	88.4	79.2	93.0	83.9	83.8
	HEAda	93.7	**52.9**	73.7	**79.0**	**78.5**	**99.5**	**90.4**	**91.2**	**92.4**	**91.1**	**80.5**	93.7	**84.6**	**84.7**

D Robustness of the OSR Gains

The main experiments show consistent improvements with the pre-specified embedding-based OSR methods, particularly KNN and Mahalanobis. We next examine how sensitive this result is to the composition of the evaluation, the choice and configuration of the OSR method, and the particular classes treated as unknown. These analyses were conducted after the main comparison and are used as robustness checks rather than for selecting the final method.

Uncertainty across settings and encoders. Because the same open-set settings are evaluated with multiple encoders, the resulting encoder–setting combinations are not independent. We therefore recompute the gains using a bootstrap that resamples both settings and encoders. The gains remain positive for KNN ($+10.14$ AUROC points; 95% CI $[4.67, 17.96]$) and Mahalanobis ($+7.79$; $[4.46, 11.67]$). In contrast, the bootstrap intervals for all four logit-based methods cross zero. This further supports that the most consistent gains from HEAda occur with embedding-based methods.

Do we need to select the OSR method? Since the best OSR score can vary across settings, we also tested whether it helps to select the scoring method using other open-set settings. In a leave-one-setting-out evaluation, however, selection provides no advantage over fixing an embedding-based method in advance. Pre-specified Mahalanobis is only 1.10 AUROC points below the per-cell oracle, followed by pre-specified KNN at 1.74 points. Selecting the method separately for each encoder gives a regret of 1.78 points, while a single method selected globally gives 1.88 points. Thus, the strong performance of HEAda does not require choosing an OSR method for each target setting.

Sensitivity to the choice of unknown classes. Our Textures protocol randomly holds out classes as unknown, raising the question of how much the result depends on the particular classes selected. We therefore repeat the experiment with five fixed class splits, training both the linear probe and HEAda with three seeds on each split. Variation across class splits is substantially larger than variation across

training seeds: 2.0–2.3× larger for HEAda and 17–26× larger for the probe (Table 10). Thus, on Textures, the choice of unknown classes contributes more than repeated optimization of a fixed split.

Table 10: **Re-drawing the unknown classes moves AUROC more than different seed.** Textures with MSP as the fixed OSR method, DINOv3-7B and SigLIP2; standard deviation (AUROC) of the per-split means across 5 splits, and the mean over splits of the standard deviation across three seeds.

Textures cell	sd across splits	sd across seeds	ratio
DINOv3-7B, HEAda	1.73	0.88	2.0×
SigLIP2, HEAda	1.27	0.56	2.3×
DINOv3-7B, probe	2.16	0.08	25.5×
SigLIP2, probe	1.20	0.07	16.8×

E Additional Analysis of Feature Geometry

The qualitative results in the main paper suggest that HEAda changes the geometry of the feature space. Here, we analyze how these changes relate to the improvements in embedding-based OSR.

Which geometric change tracks the OSR gain? HEAda changes several properties of the feature space, but it is unclear which of these changes is most closely associated with the improvement in OSR. We therefore relate three geometric changes, the separation ratio, the silhouette of the known classes, and their intra/inter-class distance ratio, to the corresponding AUROC gains of the pre-specified KNN and Mahalanobis methods (Table 11). The separation ratio, i.e., the distance of unknown samples to their nearest known class relative to the within-class distance of known samples, shows the most consistent relationship with the OSR gain. Its change correlates positively with both KNN and Mahalanobis gains across groups of settings, across all three seeds ($\rho = 0.71$, 0.66, and 0.63 for KNN), and when individual settings are left out (Table 11). In contrast, the relationship between known-class compactness and OSR gain changes across settings, so its pooled correlation combines different regimes. These correlations are descriptive: the encoder–setting–seed cells are correlated, and we do not interpret them as evidence that the geometric changes cause the improvement in OSR.

Table 11: **The change in separation ratio tracks the improvement in embedding-based OSR most consistently.** Spearman correlations between changes in each geometric statistic (HEAda minus probe) and the corresponding AUROC gain. Group columns report KNN / Mahalanobis; the final column reports the range of KNN correlations when individual settings are left out.

Predictor	ρ KNN	ρ Mah.	small	FungiTastic	ReImageNet	iNat
Δ separation	+0.66	+0.65	+0.51 / +0.36	+0.66 / +0.99	+0.94 / +0.77	+0.85 / +0.85
Δ silhouette	+0.59	+0.50	+0.49 / +0.06	+0.53 / +0.93	+0.94 / +0.94	−0.04 / +0.20
Δ log intra/inter	−0.29	−0.41	+0.26 / +0.09	−0.75 / −0.08	−0.77 / −0.66	−0.54 / −0.21

F Negative results

Our main experiments show that metric supervision is central to HEAda's gains. This left us with a natural question: how much more can we get from the structure it learns? We explored several directions that looked promising, from using the learned proxies directly at test time to introducing multiple proxies per class and explicitly learning a notion of knownness. None consistently improved the final method. These experiments were still useful: they helped us understand where the gains come from, ruled out several obvious extensions, and ultimately pushed us toward the simpler design used in the main experiments.

Can the classifier and proxy predictions be combined? HEAda gives us two ways to classify the same adapted feature: logits and the nearest proxy. We initially expected these two predictions to make different errors, making their combination useful. In practice, there is a little room for

improvement. Across 60 encoder–dataset combinations, each evaluated with 3 seeds, the classifier is correct in 83.3 % of cases and the nearest-proxy prediction in 80.3 %. An oracle that chooses the correct prediction whenever either one is right reaches only 84.6, %, leaving just 1.26 accuracy points for any fusion method. Even for DINOv3-7B (strongest encoder overall), this gap is only 0.78 points. We tested 14 fusion rules fitted on the validation data, but the best improves accuracy by only 0.07 points ($p = 0.08$, Wilcoxon test over encoder–dataset combinations; not significant after Bonferroni correction). The same pattern holds for selective prediction: using the classifier margin alone is 0.86 points better at equal coverage than requiring the classifier and proxy predictions to agree.

Can the learned proxies be used for OSR? Even if the proxy predictions do not add much to closed-set classification, the proxies could still provide useful class representatives for OSR. We therefore used the distance to the learned proxies as an OSR score and compared it with distance to class means computed from the calibration split. The simpler class means work consistently better: learned proxies reduce AUROC by 3.7 points with raw distances, and by 10.0 and 16.7 points after quantile and CDF calibration, respectively, across 60 adapted-head cells.

Would multiple proxies per class help? One possible reason learned proxies underperform class means at test time is that a single proxy may be too restrictive for classes with substantial within-class variation. We therefore extended Proxy-Anchor to use $k \in \{3, 10\}$ proxies per class, with soft assignment and a repulsion term between proxies. Across 108 runs with 6 encoders on FungiTastic-Mini, FungiTastic, and iNaturalist, using $k = 3$ changes classifier accuracy by only -0.08 points and proxy accuracy by $+0.44$ points. Increasing to $k = 10$ makes optimization substantially less stable: training diverges in 14 of 18 FungiTastic runs for encoders below 3 072 dimensions. We tried several ways to stabilize training, including a normalized repulsion term, annealed temperature, k-means initialization, and a separate projection, but none produced a consistent improvement over the single-proxy formulation.

Would refitting the proxies help? Another possibility is that the proxies learned jointly with the adaptation head are not the best representatives of the final embedding space. We therefore froze the trained head and re-optimized only the proxies. The results are mixed: across 54 runs, refitting improves accuracy by 0.91 points on FungiTastic, has no effect on FungiTastic-Mini, and reduces accuracy by 0.38 points on iNaturalist. The improvement is therefore specific to FungiTastic rather than a consistent benefit of refitting.

Can we explicitly learn knownness? We also explored a more direct approach to OSR: encoding class identity in the feature angle and "knownness" in its norm. We trained a radial ArcFace head with six different sources of exposed outliers, across 108 runs with DINOv3-7B and BioCLIP 2. This approach never outperforms all twelve standard OSR methods. Even its best AUROC, 0.864 when using the dataset's own validation unknowns for exposure, remains below MaxLogit without outlier exposure (0.896). More importantly, the result is highly sensitive to where the exposed outliers come from. Using fungi from a different collection (iNaturalist) drops AUROC to 0.212, compared with 0.481 without exposure. This suggests that feature norm can capture differences in acquisition conditions rather than providing a reliable signal of semantic novelty. These experiments predate our final evaluation protocol and were not recomputed, so we treat them as diagnostic rather than part of the main empirical evidence.

Does logit calibration explain the difference? Finally, we checked whether calibration could explain the behavior of the classifier logits. The HEAda logits are slightly under-confident, with an optimal temperature of $T = 0.895$, and temperature scaling reduces ECE from 0.088 to 0.021. In contrast, the Proxy-Anchor+ArcFace head (9a) is strongly over-confident, with $T = 4.5$. Thus, the two heads differ substantially in calibration, but correcting their temperature does not change their closed-set decisions.

Taken together, these experiments pushed the HEAda design toward the simpler one: metric supervision shapes the embedding space during training, while standard OSR methods remain responsible for detecting unknowns at test time.