VOCABULARY-INCOMPLETE UNSUPERVISED VISION-LANGUAGE MODEL ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

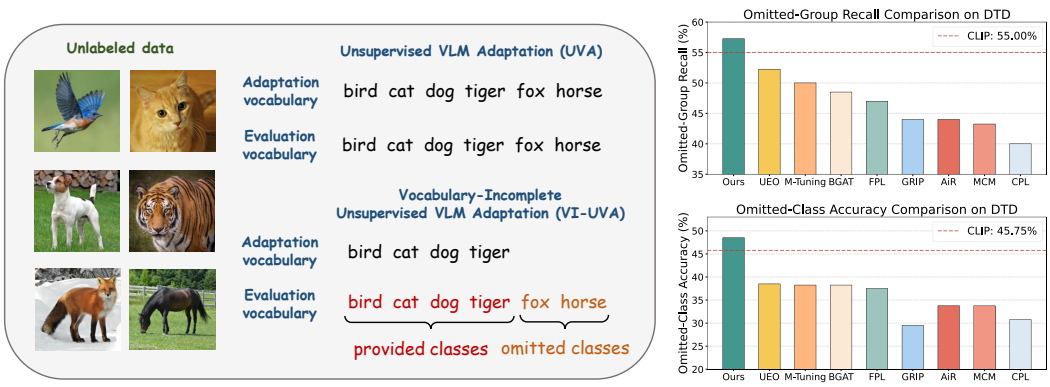
In certain real-world scenarios, obtaining labeled data is excessively expensive, motivating increasing interest in adapting models without annotations. Vision-language models (VLMs) provide pseudo-labels for unlabeled images through image-text similarity, enabling adaptation without manual labeling. The previous unsupervised vision-language model adaption methods assume access to an adaptation vocabulary containing all class names. However, a complete vocabulary is difficult to construct in real-world scenarios, as unlabeled data may contain classes that are overlooked or fall outside the current task scope. Thus, we propose **V**ocabulary-**I**ncomplete **U**nsupervised **V**ision-language model **A**daptation (VI-UVA), where only a subset of class names are available during adaptation. The vocabulary incompleteness induces an adaptation bias toward provided classes, which degrades omitted-class recognition. To address this issue, we propose **L**abel **S**pace **R**econstruction (LSR), which reconstructs a latent label space from the unlabeled data and incorporates it into prompt tuning, thereby extending adaptation beyond the incomplete vocabulary and mitigating the adaptation bias. Furthermore, we derive an omitted-class semantic discrepancy bound for LSR and identify the key factors affecting the bound. Extensive experiments across eight benchmarks demonstrate that LSR outperforms existing state-of-the-art methods. ***Code is available in the supplementary materials.***

1 INTRODUCTION

Vision-language models (VLMs) exhibit strong open-vocabulary generalization capabilities but often require fine-tuning to further improve performance on specific downstream tasks (Peleg et al., 2025; Chuang et al., 2025). Nevertheless, acquiring labeled data for fine-tuning can be excessively expensive in certain real-world scenarios (Wang et al., 2025b; Ren et al., 2025), driving increasing interest in adapting VLMs without manual annotations. **U**nsupervised **V**ision-language model **A**daptation (**UVA**) (Huang et al., 2022) aims to improve the downstream performance of pre-trained VLMs with unlabeled data. Existing UVA methods (Menghini et al., 2023; Zhang et al., 2024a) typically construct textual prototypes from the adaptation vocabulary and assign pseudo-labels to unlabeled data based on image-text similarity. High-confidence pseudo-labeled samples are then selected to fine-tune the VLM for the target task. While effective in UVA, these methods rely on the assumption that the adaptation vocabulary contains all class names present in the unlabeled data.

However, this assumption may not hold in real-world scenarios, where unlabeled data contains classes beyond the scope of the current adaptation task. For example, when researchers adapt a VLM to monitor target species, the adaptation vocabulary may omit some species present in the wildlife images because they fall outside the current task scope or are overlooked during task specification (Zhu et al., 2022; Gabeff et al., 2024; Chen et al., 2025). As monitoring expands beyond target species, researchers need to apply the adapted model to recognize other species. The species names previously omitted can be added to the evaluation vocabulary as textual queries (Gabeff et al., 2024; Stevens et al., 2024). Motivated by this scenario, we propose **V**ocabulary-**I**ncomplete **U**nsupervised **V**ision-language model **A**daptation (**VI-UVA**). In this setting, only a subset of class names (i.e., provided-class names) is available during adaptation, while the complete vocabulary is only available at the evaluation stage. Fig. 1a shows the difference between UVA and VI-UVA.

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.



(a) (b)

Figure 1: (a) Comparison between unsupervised VLM adaptation (UVA) and vocabulary-incomplete unsupervised VLM adaptation (VI-UVA). During adaptation, all class names are available for UVA, while only a subset of class names is available for VI-UVA. During evaluation, UVA and VI-UVA are both tested on the complete vocabulary including provided-class names and omitted-class names. (b) Omitted-group recall measures the proportion of omitted-class samples assigned to any omitted class, which indicates how well the model avoids vocabulary overfitting. On the DTD dataset, our method outperforms the best adaptation method by 5.0 and 10.0 percentage points in omitted-group recall and omitted-class accuracy, respectively.

Existing UVA methods rely on pseudo-labeling to adapt VLMs. However, under the proposed setting, pseudo-labeling may assign omitted-class images to provided classes, while training exclusively with the incomplete vocabulary drives the learned prompts toward the provided classes, inducing an **adaptation bias**. This bias leads to vocabulary overfitting (i.e., the model tends to classify omitted-class samples as provided classes) and degrades omitted-class accuracy (Fig. 1b). Thus, the main issue is to improve provided-class recognition and preserve omitted-class recognition, even though the information of omitted classes are unavailable during adaptation.

To address this issue, we propose **L**abel **S**pace **R**econstruction (LSR). The main idea is to exploit the latent semantic structures in unlabeled data, which contains images from both provided and omitted classes, offering information beyond the adaptation vocabulary. To extract this semantic information, we train a lightweight **C**ross-**M**odal **S**emantic **P**rojector (CMSP) that maps visual representations into the textual embedding space. We apply CMSP to all unlabeled images and cluster the generated textual embeddings to obtain semantic anchors. Each anchor is associated with a corresponding pseudo-token sequence, constructing a pseudo-vocabulary that encodes latent semantics. By incorporating the pseudo-vocabulary into prompt tuning, LSR prevents the prompts from being tuned only to the provided classes. This helps mitigate the adaptation bias and preserve omitted-class recognition (Fig. 1b). Finally, we derive an omitted-class semantic discrepancy bound for LSR, characterizing how projection quality, cross-modal recomposition, and representation approximation jointly govern the bound.

In summary, the main contributions of this work are as follows.

- For the first time, we propose **V**ocabulary-**I**ncomplete **U**nsupervised **V**ision-language **M**odel **A**daptation (VI-UVA), a practical yet challenging setting in which only a subset of class names is available during adaptation. The vocabulary incompleteness induces an adaptation bias, which leads to vocabulary overfitting and degrades omitted-class recognition.
- We propose **L**abel **S**pace **R**econstruction (LSR), which reconstructs a latent label space from the unlabeled data and incorporates it into prompt tuning, thereby extending fine-tuning beyond the adaptation vocabulary and mitigating the adaptation bias.
- We establish an omitted-class semantic discrepancy bound for LSR, which is jointly determined by projection quality, cross-modal recomposition, and representation approximation.
- Extensive experiments on eight commonly used benchmarks at different omission ratios demonstrate that our method achieves state-of-the-art performance.

2 Related Work

2.1 Vision-Language Models

Vision-Language Models (VLMs) (Radford et al., 2021; Li et al., 2021; 2022; Xiao et al., 2024) learn cross-modal representations from large-scale image-text pairs. Among them, CLIP (Radford et al., 2021) consists of an image encoder and a text encoder that project images and textual descriptions into a shared embedding space. This alignment enables classification through textual descriptions without additional training on the target task. Specifically, class names are inserted into predefined prompt templates, such as *a photo of a [class]*, and encoded as textual prototypes. Each input image is then assigned to the class whose textual prototype has the highest similarity to its visual representation. Despite their strong zero-shot capabilities, VLMs suffer substantial performance degradation under domain shifts, motivating adaptation to downstream tasks. Existing studies adapt VLMs by optimizing prompts with labeled data (Zhou et al., 2022b;a; Zhang et al., 2024b). In this paper, we focus on unsupervised VLM adaptation, thereby eliminating the need for costly annotations.

2.2 Unsupervised Vision-Language Model Adaptation

In certain real-world scenarios, acquiring even a small number of annotations can be excessively expensive (Zhang et al., 2024a; Wang et al., 2025b; Ren et al., 2025). This limitation has motivated increasing interest in unsupervised VLM adaptation, where zero-shot predictions from VLMs are used as pseudo-labels for prompt tuning. Representative methods such as FPL and GRIP (Menghini et al., 2023) select the top-k highest-confidence samples for each class, with GRIP further refining the pseudo-labels and progressively increasing k during adaptation. Subsequent studies extend this framework by introducing partial labels (Zhang et al., 2024a), mitigating prediction bias through calibrated sample selection (Wang et al., 2025b), or incorporating auxiliary supervision from generated images (Ren et al., 2025). However, these methods implicitly assume that the adaptation vocabulary is complete. In contrast, we investigate a vocabulary-incomplete setting in which only a subset of class names is available during adaptation.

2.3 Related Learning Paradigms

Several learning paradigms relax the closed-set assumption and address different forms of openness. Open-set recognition (Bendale & Boult, 2016) and out-of-distribution detection (Hendrycks & Gimpel, 2016) reject samples outside the known classes or training distribution, whereas classes omitted from the adaptation vocabulary in VI-UVA still require fine-grained classification. Generalized category discovery (Vaze et al., 2022; Wang et al., 2025a) classifies known classes and discovers novel classes by clustering unlabeled data, typically with access to labeled data from known classes, while VI-UVA uses only unlabeled data and an incomplete vocabulary to classify all classes. A detailed comparison of VI-UVA with these learning paradigms is shown in Appendix A. Overall, VI-UVA seeks to perform classification over all classes while relying on weaker supervision, making it more challenging.

3 Proposed Method

Overview. In this paper, we propose **L**abel **S**pace **R**econstruction (LSR) for vocabulary-incomplete unsupervised vision-language model adaptation. LSR consists of three stages, including cross-modal semantic projection (Sec. 3.1), latent label space reconstruction (Sec. 3.2), and reconstruction space adaptation (Sec. 3.3). The overview and pseudo-code of LSR are detailed in Fig. 5 and Algorithm 1, respectively.

Problem Setup. Let $\mathcal{D}_{\mathrm{U}} = \{x_i\}_{i=1}^{N}$ denote an unlabeled dataset containing N samples and $\mathcal{C}_t = \{c_j\}_{j=1}^{C}$ denote the set of all class names. Each sample in $\mathcal{D}_{\mathrm{U}}$ belongs to a class whose name is included in $\mathcal{C}_t$. We partition $\mathcal{C}_t$ into two disjoint subsets,

$$\mathcal{C}_t = \mathcal{C}_p \cup \mathcal{C}_o, \quad \mathcal{C}_p \cap \mathcal{C}_o = \emptyset, \tag{1}$$

where $\mathcal{C}_p$ and $\mathcal{C}_o$ denote the sets of provided- and omitted-class names, respectively. Specifically, the class names in $\mathcal{C}_p$ constitute the adaptation vocabulary $\mathcal{V}_p$, whereas those in $\mathcal{C}_o$ are unavailable during adaptation, even though their images are present in $\mathcal{D}_{\mathrm{U}}$.

Given $\mathcal{D}_{\mathrm{U}}$ and $\mathcal{V}_p$, we aim to adapt the VLM without annotations or the class names in $\mathcal{C}_o$. The adapted model is evaluated on $\mathcal{C}_t$, with the goal of maximizing recognition performance across both provided and omitted classes.

3.1 Cross-Modal Semantic Projection

The adaptation vocabulary contains only a subset of class names. Pseudo-labeling with this incomplete vocabulary can assign omitted-class images to provided classes, which introduces biased supervision and encourages the model to overfit the adaptation vocabulary. Motivated by Wu et al. (2023), we introduce the **C**ross-**M**odal **S**emantic **P**rojector (CMSP), a lightweight linear mapping that transfers visual representations into the textual embedding space, providing a basis for reconstructing the latent semantic anchors associated with omitted classes.

Denote the image encoder and text encoder of VLMs as $\mathcal{I}(\cdot)$ and $\mathcal{T}(\cdot)$, respectively. We extract a visual embedding $\mathbf{x}_i \in \mathbb{R}^d$ of each $x_i \in \mathcal{D}_{\mathrm{U}}$ through $\mathcal{I}(\cdot)$ and extract a textual embedding $\mathbf{c}_j \in \mathbb{R}^d$ of each $c_j \in \mathcal{V}_p$ through $\mathcal{T}(\cdot)$. Based on image-text similarity, these textual embeddings are used to generate pseudo-labels for unlabeled images. Following prior work (Menghini et al., 2023), we select the same number of highest-confidence images for each provided class to construct a balanced pseudo-labeled dataset $\mathcal{D}_{\mathrm{P}} = \{(x_m, p_m)\}_{m=1}^{M}$, where $p_m \in \{1, \cdots, |\mathcal{V}_p|\}$ denotes the pseudo-label assigned to image x_m. Appendix F.1 provides the details of pseudo-labeled dataset generation. $\mathcal{D}_{\mathrm{P}}$ is then used to train the cross-modal semantic projector.

The cross-modal semantic projector, parameterized by ϕ, maps the visual embedding $\mathbf{x}_m \in \mathbb{R}^d$ of each pseudo-labeled image x_m into a flattened token-level representation $\mathbf{h}_m \in \mathbb{R}^{Ld}$, where L is the predefined number of pseudo-tokens. We reshape $\mathbf{h}_m$ into a pseudo-token sequence $\mathbf{H}_m \in \mathbb{R}^{L \times d}$ and feed it into the frozen text encoder $\mathcal{T}(\cdot)$ to obtain the generated textual embedding $\mathbf{t}_m \in \mathbb{R}^d$. The complete projection process can be expressed as:

$$\mathbf{t}_m = \mathcal{T}\left(\mathcal{R}_{L \times d}\left(f_\phi(\mathcal{I}(x_m))\right)\right), \tag{2}$$

where $\mathcal{R}_{L \times d}$ represents the reshape operation and f_ϕ denotes the CMSP. The training objective of the CMSP comprises a feature regression loss and a classification loss.

Feature regression loss. To encourage the generated textual embedding $\mathbf{t}_m$ to preserve the category-level semantics, we minimize its feature distance to the corresponding provided-class prototype. Let $\mathbf{c}_{p_m}$ denote the textual embedding of the pseudo-label p_m. Given a mini-batch of B pseudo-labeled samples, the feature regression loss is defined as:

$$\mathcal{L}_{feat} = \frac{1}{Bd} \sum_{m=1}^{B} \left\| \mathbf{t}_m - \mathbf{c}_{p_m} \right\|_1, \tag{3}$$

where d is the embedding dimension. This objective directly encourages the generated textual embeddings to align closely with their corresponding class prototypes.

Classification loss. Feature regression loss alone does not directly optimize the relative similarities between a generated textual embedding and different provided-class prototypes. Therefore, we calculate its similarities to all provided-class prototypes and apply a classification loss:

$$\mathcal{L}_{cls} = -\frac{1}{B} \sum_{m=1}^{B} \log \frac{\exp\left(\mathbf{t}_m^\top \mathbf{c}_{p_m}\right)}{\sum_{j=1}^{|\mathcal{C}_p|} \exp\left(\mathbf{t}_m^\top \mathbf{c}_j\right)}, \tag{4}$$

where $|\mathcal{C}_p|$ represents the number of provided classes. This loss encourages each generated textual embedding to be more similar to its corresponding class prototype than to the prototypes of other provided classes. Finally, the overall objective for training the CMSP is formulated as:

$$\mathcal{L}_{\mathrm{CMSP}} = \mathcal{L}_{feat} + \alpha \mathcal{L}_{cls}, \tag{5}$$

where α is set to 0.1 in our implementation. During training, the image and text encoders remain frozen, and only the parameters ϕ of the CMSP are optimized.

3.2 Latent Label Space Reconstruction

By applying the projection process in Eq. 2 to all images in $\mathcal{D}_{\mathrm{U}}$, we obtain the generated textual embeddings $\mathbf{G} = [\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_N]^\top \in \mathbb{R}^{N \times d}$ derived from both provided- and omitted-class images. Thus, we exploit these embeddings to reconstruct a latent label space.

Latent semantic clustering. Since the information of omitted classes are unavailable during adaptation, we adopt an over-clustering strategy to allocate additional semantic capacity beyond the provided classes. Specifically, the number of clusters is set to $2|\mathcal{C}_p|$, which is twice the number of provided classes. We perform K-means clustering on the generated textual embeddings $\mathbf{G}$ and the resulting candidate latent semantic anchors are denoted by $\mathcal{M} = \{\boldsymbol{\mu}_k\}_{k=1}^{2|\mathcal{C}_p|}$, where $\boldsymbol{\mu}_k \in \mathbb{R}^d$ represents the k-th cluster center. These cluster centers serve as data-driven semantic anchors and reserve additional capability to represent the concepts beyond the adaptation vocabulary. Importantly, they are not required to recover the ground-truth class names or correspond one-to-one with the underlying classes.

Over-clustering may produce unreliable anchors associated with outliers or noise. We assess each candidate anchor by the number of unlabeled images assigned to it and discard those supported by very few images. This filtering procedure aims to reduce the influence of outliers or noise, while retaining the dominant semantic structures present in the unlabeled data. The reconstructed latent label space is consequently denoted as $\mathcal{M}_{ret} = \{\boldsymbol{\mu}_k\}_{k=1}^{r}$, where r represents the number of retained anchors. Appendix F.2 provides the details of the filtering procedure.

3.3 Reconstructed Space Adaptation

Since the retained anchors cannot be directly used as text encoder inputs, we associate them with pseudo-token sequences generated by CMSP for prompt tuning. Recall that CMSP maps the visual representation of each unlabeled image x_i into a pseudo-token sequence $\mathbf{H}_i \in \mathbb{R}^{L\times d}$ and the text encoder generates the corresponding textual embedding $\mathbf{t}_i = \mathcal{T}(\mathbf{H}_i)$. For each retained anchor $\boldsymbol{\mu}_k \in \mathcal{M}_{ret}$, we retrieve its most similar generated textual embedding $\mathbf{t}_{i_k^*}$, where

$$i_k^* = \underset{i\in\{1,\dots,N\}}{\arg\max} \frac{\boldsymbol{\mu}_k^\top \mathbf{t}_i}{\|\boldsymbol{\mu}_k\|_2 \|\mathbf{t}_i\|_2}. \tag{6}$$

Consequently, the pseudo-token sequence associated with the k-th anchor is denoted as $\mathbf{H}_{i_k^*}$. The reconstructed pseudo-vocabulary and corresponding textual representations are denoted by:

$$\mathcal{V}_{\mathrm{R}} = \left\{\mathbf{H}_{i_k^*}\right\}_{k=1}^{r}, \quad \mathbf{T}_{\mathrm{R}} = \left\{\mathbf{t}_{i_k^*}\right\}_{k=1}^{r}, \tag{7}$$

where r represents the number of retained semantic anchors. We exploit $\mathbf{T}_{\mathrm{R}}$ to construct a balanced pseudo-labeled dataset $\mathcal{D}_{\mathrm{R}} = \{(x_q, p_q)\}_{q=1}^{Q}$ following the same procedure described in the Appendix F.1, where Q denotes the number of selected images and $p_q \in \{1,\dots,r\}$ represents the latent pseudo-label of the image x_q.

Then, we perform textual prompt tuning (Zhou et al., 2022b) with $\mathcal{V}_{\mathrm{R}}$ and $\mathcal{D}_{\mathrm{R}}$. Let $\mathbf{V}_\theta = [\mathbf{v}_1, \cdots, \mathbf{v}_S]^\top \in \mathbb{R}^{S\times d}$ denote the shared learnable context tokens, where S is the number of shared tokens. For the k-th pseudo-token sequence in $\mathcal{V}_{\mathrm{R}}$, its prompt-conditioned textual prototype is computed as:

$$\mathbf{z}_k = \mathcal{T}\left(\left[\mathbf{V}_\theta; \mathbf{H}_{i_k^*}\right]\right), \tag{8}$$

where $[\cdot;\cdot]$ denotes token-level concatenation. Given a mini-batch of B pseudo-labeled samples (x_q, p_q) from $\mathcal{D}_{\mathrm{R}}$, the cross-entropy loss for optimizing $\mathbf{V}_\theta$ is defined as:

$$\mathcal{L}_{\mathrm{R}} = -\frac{1}{B}\sum_{q=1}^{B} \log \frac{\exp\left(\mathrm{sim}\left(\mathcal{I}(x_q), \mathbf{z}_{p_q}\right)/\tau\right)}{\sum_{\ell=1}^{r}\exp\left(\mathrm{sim}\left(\mathcal{I}(x_q), \mathbf{z}_\ell\right)/\tau\right)}, \tag{9}$$

where τ is the temperature parameter and $\mathrm{sim}(\cdot,\cdot)$ represents cosine similarity. This objective updates $\mathbf{V}_\theta$ using supervision from the pseudo-vocabulary $\mathcal{V}_{\mathrm{R}}$, which incorporates semantic structures derived from both provided and omitted classes, thereby helping preserve omitted-class recognition. To further enhance provided-class recognition, we also optimize $\mathbf{V}_\theta$ with the balanced pseudo-labeled dataset $\mathcal{D}_{\mathrm{P}}$. For each provided-class name $c_j \in \mathcal{V}_p$, we construct a prompt-conditioned textual prototype as:

$$\widetilde{\mathbf{c}}_j = \mathcal{T}\left(\left[\mathbf{V}_\theta; \mathrm{Token}(c_j)\right]\right), \tag{10}$$

where $\mathrm{Token}(c_j)$ denotes the token embeddings of the class name c_j. Given a mini-batch of B pseudo-labeled samples (x_m, p_m) from $\mathcal{D}_{\mathrm{P}}$, the cross-entropy loss is:

$$\mathcal{L}_{\mathrm{P}} = -\frac{1}{B}\sum_{m=1}^{B} \log \frac{\exp\left(\mathrm{sim}\left(\mathcal{I}(x_m), \widetilde{\mathbf{c}}_{p_m}\right)/\tau\right)}{\sum_{\ell=1}^{|\mathcal{C}_p|}\exp\left(\mathrm{sim}\left(\mathcal{I}(x_m), \widetilde{\mathbf{c}}_\ell\right)/\tau\right)}. \tag{11}$$

The overall objective for textual prompt tuning is:

$$\mathcal{L}_{\text{all}} = \lambda \mathcal{L}_{\text{R}} + \mathcal{L}_{\text{P}}, \tag{12}$$

where λ balances supervision from the reconstructed latent label space and the provided classes. During prompt tuning, the image encoder, text encoder, CMSP, and pseudo-vocabulary $\mathcal{V}_{\text{R}}$ remain frozen, while only the shared context tokens $\mathbf{V}_\theta$ are optimized. At inference time, we combine the shared context tokens with each class name in $\mathcal{C}_t$ to recognize both provided and omitted classes.

3.4 Theoretical Analysis

We establish an omitted-class semantic discrepancy bound for LSR, identifying how projection quality, cross-modal recomposition, and representation approximation jointly govern the bound.

Theorem 1 (Omitted-Class Semantic Discrepancy Bound). *Let $\mathcal{F}(\mathbf{x}) = \mathcal{T}\left(\mathcal{R}_{L\times d}\left(f_\phi(\mathbf{x})\right)\right)$ denote the complete visual-to-textual mapping. Suppose that $\mathcal{F}$ and its Jacobian are Lipschitz continuous on a convex domain containing the relevant visual representations. Then,*

$$\mathcal{E}_o^{\text{rec}} \leq \frac{1}{2}\left(\varepsilon_{\text{proj}} + \mathfrak{b}_o + \mathfrak{r}_o\right)^2, \tag{13}$$

where $\mathcal{E}_o^{\text{rec}}$ denotes the average semantic discrepancy between the nearest retrieved textual representations and corresponding omitted-class prototypes, $\varepsilon_{\text{proj}}$ measures the error of cross-modal semantic projection, $\mathfrak{b}_o$ measures the discrepancy of recomposing omitted-class visual and textual semantics, and $\mathfrak{r}_o$ measures the approximation error of the generated textual representations to their nearest retrieved textual representations.

Theorem 1 reveals three factors affecting omitted-class semantic discrepancy. Accurate projection, lower cross-modal recomposition discrepancy and representation approximation error jointly lead to a tighter bound. The complete proof and further analysis of these factors are provided in Appendix C.

4 Experiments

To demonstrate the effectiveness of the proposed method, we conduct experiments across eight datasets and compare with ten SOTA methods. In addition, we conduct ablation studies and analytical experiments to explain why our method is effective.

4.1 Experimental Settings

Datasets and evaluations. We conduct experiments on eight image classification datasets from diverse domains, including DTD (DTD.) (Cimpoi et al., 2014), FGVC-Aircraft (FA.) (Maji et al., 2013), CUB (CUB.) (Wah et al., 2011), Flowers102 (Flw.) (Nilsback & Zisserman, 2008), OxfordPets (Pets.) (Parkhi et al., 2012), StanfordCars (Cars.) (Krause et al., 2013), RESISC45 (RES.) (Cheng et al., 2017), and CIFAR100 (CIFAR.) (Krizhevsky, 2009). For each dataset, we partition the set of all class names $\mathcal{C}_t$ into provided and omitted subsets. We define the **omission ratio** as $\gamma = |\mathcal{C}_o|/|\mathcal{C}_t|$ and evaluate multiple values of γ to simulate different degrees of vocabulary incompleteness. We evaluate model performance using three metrics, including all-class accuracy, omitted-class accuracy, and omitted-group recall. Further details can be found in Appendix D.

Baselines. We compare our method with ten SOTA methods, namely, CLIP (Radford et al., 2021), FPL (Menghini et al., 2023), GRIP (Menghini et al., 2023), CPL (Zhang et al., 2024a), UEO (Liang et al., 2024), AiR (Ren et al., 2025), CAP (Wang et al., 2025b), MCM (Ming et al., 2022), M-Tuning (Liao et al., 2025) and BGAT (Li et al., 2026). Appendix E describes the details about these methods. For a fair comparison, all adaptation methods use the same pre-trained CLIP backbone and employ textual prompt tuning (Zhou et al., 2022b). Appendix F.3 presents the implementation details.

4.2 Main Results

Tables 1 and 2 compare the all-class accuracy of different methods across eight datasets at omission ratios of $\gamma = 0.2$ and $\gamma = 0.3$, respectively. Each result is reported as the mean and standard deviation over three random seeds for optimization (1, 2, and 3). Our method consistently achieves the highest

accuracy on all eight datasets. At $\gamma = 0.2$ and $\gamma = 0.3$, LSR outperforms the strongest baseline by an average of **2.47** and **2.57** percentage points, respectively. Specifically, LSR surpasses the best baseline on RES by **3.74** percentage points at $\gamma = 0.3$. These results demonstrate the effectiveness of LSR across different datasets and omission ratios, suggesting that the reconstructed latent label space complements the restricted supervision provided by the adaptation vocabulary. We provide more main results in Appendix G.1.

Table 1: Comparison results of all-class accuracy (%) at $\gamma = 0.2$. Values are expressed as mean_{std}. The highest accuracies are bold.

Methods	Flw.	FA.	CUB.	DTD.	Pets.	Cars.	RES.	CIFAR.	Avg.
CLIP	$63.63_{0.00}$	$17.64_{0.00}$	$51.77_{0.00}$	$43.19_{0.00}$	$84.51_{0.00}$	$58.93_{0.00}$	$54.48_{0.00}$	$64.19_{0.00}$	$54.79_{0.00}$
FPL	$65.99_{0.70}$	$17.11_{0.46}$	$51.09_{0.58}$	$44.29_{0.67}$	$87.95_{0.61}$	$57.09_{0.71}$	$58.08_{0.68}$	$65.05_{0.55}$	$55.83_{0.62}$
GRIP	$60.83_{0.25}$	$14.68_{0.37}$	$42.21_{0.72}$	$44.24_{2.00}$	$79.87_{0.62}$	$50.53_{0.35}$	$62.57_{1.38}$	$65.02_{1.17}$	$52.49_{0.86}$
CPL	$63.50_{1.90}$	$14.97_{0.65}$	$46.82_{0.71}$	$45.53_{0.33}$	$77.56_{1.10}$	$47.17_{0.57}$	$64.06_{0.92}$	$62.17_{0.53}$	$52.72_{0.84}$
CAP	$65.37_{1.60}$	$15.74_{0.91}$	$49.18_{0.75}$	$45.85_{2.25}$	$80.43_{4.37}$	$53.92_{0.24}$	$65.57_{1.32}$	$65.25_{0.04}$	$55.16_{1.44}$
AiR	$60.61_{1.13}$	$14.98_{0.11}$	$41.62_{0.72}$	$45.87_{1.42}$	$78.88_{2.17}$	$49.84_{0.11}$	$63.72_{1.02}$	$65.33_{0.66}$	$52.61_{0.92}$
MCM	$61.81_{0.35}$	$16.19_{1.23}$	$45.53_{1.14}$	$39.73_{0.97}$	$79.47_{1.74}$	$51.53_{0.32}$	$53.11_{0.65}$	$63.46_{0.45}$	$51.35_{0.86}$
UEO	$61.74_{0.92}$	$8.29_{2.95}$	$44.52_{0.18}$	$42.86_{0.20}$	$82.73_{4.11}$	$52.39_{1.24}$	$64.94_{1.13}$	$58.79_{0.97}$	$52.03_{1.46}$
M-Tuning	$66.75_{0.29}$	$17.31_{0.31}$	$50.79_{0.23}$	$44.70_{0.24}$	$88.06_{0.55}$	$57.26_{0.12}$	$58.11_{0.42}$	$65.41_{0.33}$	$56.05_{0.31}$
BGAT	$58.72_{1.41}$	$18.22_{0.77}$	$48.24_{0.70}$	$40.05_{0.16}$	$82.60_{0.65}$	$55.87_{0.34}$	$54.47_{0.42}$	$64.54_{0.07}$	$52.84_{0.57}$
LSR (Ours)	$\mathbf{66.89}_{0.60}$	$\mathbf{19.34}_{0.41}$	$\mathbf{53.27}_{0.22}$	$\mathbf{47.02}_{0.54}$	$\mathbf{88.49}_{0.60}$	$\mathbf{60.15}_{0.34}$	$\mathbf{65.82}_{0.35}$	$\mathbf{67.14}_{0.53}$	$\mathbf{58.52}_{0.45}$

Table 2: Comparison results of all-class accuracy (%) at $\gamma = 0.3$. Values are expressed as mean_{std}. The highest accuracies are bold.

Methods	Flw.	FA.	CUB.	DTD.	Pets.	Cars.	RES.	CIFAR.	Avg.
CLIP	$63.63_{0.00}$	$17.64_{0.00}$	$51.77_{0.00}$	$43.19_{0.00}$	$84.51_{0.00}$	$58.93_{0.00}$	$54.48_{0.00}$	$64.19_{0.00}$	$54.79_{0.00}$
FPL	$63.29_{0.62}$	$17.23_{0.10}$	$50.70_{0.52}$	$43.14_{0.56}$	$86.80_{0.36}$	$55.52_{0.91}$	$57.14_{1.48}$	$64.88_{0.44}$	$54.84_{0.62}$
GRIP	$53.58_{2.79}$	$14.65_{0.41}$	$40.66_{1.31}$	$42.29_{0.74}$	$75.66_{1.11}$	$49.50_{0.87}$	$59.36_{1.73}$	$63.06_{1.07}$	$49.85_{1.25}$
CPL	$56.59_{0.53}$	$15.12_{0.80}$	$45.10_{0.51}$	$41.72_{1.54}$	$74.58_{1.75}$	$45.02_{0.48}$	$59.32_{1.15}$	$61.21_{1.48}$	$49.83_{1.03}$
CAP	$56.23_{0.51}$	$14.88_{0.98}$	$45.64_{0.38}$	$43.09_{1.03}$	$81.30_{3.24}$	$53.64_{0.74}$	$61.52_{0.36}$	$64.80_{0.33}$	$52.64_{0.95}$
AiR	$57.07_{1.61}$	$14.11_{1.35}$	$40.32_{1.02}$	$42.52_{1.45}$	$81.78_{1.19}$	$50.18_{1.30}$	$59.70_{0.35}$	$62.69_{0.46}$	$51.05_{1.09}$
MCM	$58.76_{0.48}$	$14.44_{2.23}$	$44.99_{1.12}$	$38.10_{0.83}$	$81.16_{1.10}$	$49.88_{1.02}$	$51.81_{0.19}$	$63.70_{0.19}$	$50.36_{0.90}$
UEO	$60.19_{1.82}$	$7.23_{1.80}$	$43.66_{2.05}$	$40.60_{1.55}$	$81.71_{0.88}$	$49.21_{2.33}$	$60.78_{1.82}$	$53.76_{1.38}$	$49.64_{1.70}$
M-Tuning	$64.16_{0.79}$	$17.02_{0.68}$	$50.46_{0.39}$	$43.74_{0.55}$	$86.89_{0.26}$	$55.22_{0.21}$	$58.65_{0.95}$	$65.07_{0.47}$	$55.15_{0.54}$
BGAT	$57.72_{1.33}$	$17.15_{0.06}$	$47.27_{0.87}$	$39.15_{1.19}$	$81.87_{0.41}$	$55.50_{0.78}$	$51.87_{0.64}$	$64.80_{0.25}$	$51.92_{0.69}$
LSR (Ours)	$\mathbf{65.50}_{0.42}$	$\mathbf{18.92}_{0.18}$	$\mathbf{53.00}_{0.50}$	$\mathbf{45.32}_{0.52}$	$\mathbf{87.53}_{0.33}$	$\mathbf{59.68}_{0.29}$	$\mathbf{65.26}_{0.41}$	$\mathbf{66.53}_{0.14}$	$\mathbf{57.72}_{0.35}$

4.3 Ablation Studies

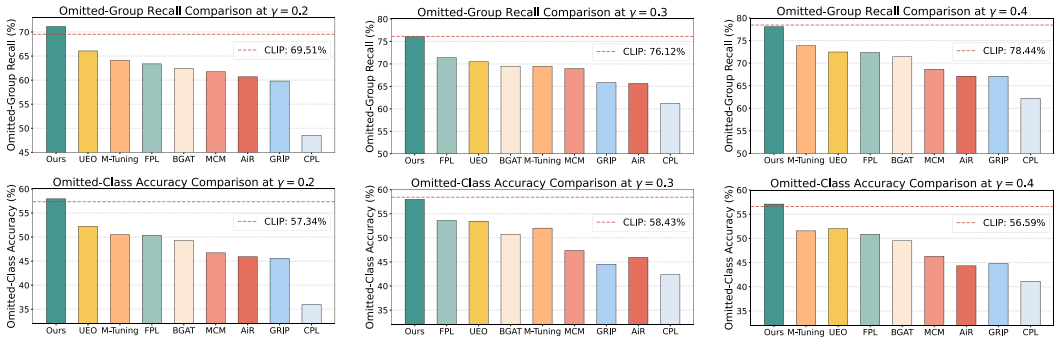
Table 3 examines the contributions of the key components in our method at $\gamma = 0.4$. We report all-class accuracy of different ablation variants, with each result expressed as mean_{std}. The variant trained solely with $\mathcal{L}_{\text{P}}$ improves average accuracy slightly, from 55.16% to 55.49%. This limited improvement stems from supervision restricted to the adaptation vocabulary while neglecting to preserve omitted-class recognition. Incorporating $\mathcal{L}_{\text{R}}$ extends supervision to the latent label space reconstructed from CMSP-generated representations. Thus, we evaluate the contribution of each loss used to train the CMSP. Training the CMSP with only $\mathcal{L}_{\text{cls}}$ or $\mathcal{L}_{\text{feat}}$ achieves average accuracies of 56.97% and 56.69%, respectively. Both variants outperform prompt tuning with $\mathcal{L}_{\text{P}}$ alone, supporting the benefit of reconstruction-based supervision. Combining both losses improves average accuracy to **57.54**% and yields the best results on all six datasets. This suggests that $\mathcal{L}_{\text{cls}}$ and $\mathcal{L}_{\text{feat}}$ complement each other, enabling more effective latent label space reconstruction. Appendix G.2 also compares the omitted-class accuracy between variants with and without $\mathcal{L}_{\text{R}}$.

Table 3: Comparison of all-class accuracy (%) for ablation variants at $\gamma = 0.4$. Values are expressed as mean_{std}. The highest accuracies are bold.

$\mathcal{L}_{\text{P}}$	$\mathcal{L}_{\text{R}}$	$\mathcal{L}_{\text{feat}}$	$\mathcal{L}_{\text{cls}}$	Flw.	FA.	CUB.	Pets.	Cars.	RES.	Avg.
✗	✗	✗	✗	$63.63_{0.00}$	$17.64_{0.00}$	$51.77_{0.00}$	$84.51_{0.00}$	$58.93_{0.00}$	$54.48_{0.00}$	$55.16_{0.00}$
✓	✗	✗	✗	$62.45_{0.65}$	$17.16_{0.56}$	$50.36_{0.99}$	$83.19_{0.89}$	$57.07_{0.47}$	$62.72_{0.55}$	$55.49_{0.69}$
✓	✓	✗	✓	$63.79_{1.16}$	$18.33_{0.26}$	$51.70_{0.48}$	$85.75_{0.60}$	$58.78_{0.13}$	$63.49_{0.30}$	$56.97_{0.49}$
✓	✓	✓	✗	$63.12_{0.49}$	$18.07_{0.65}$	$52.02_{0.36}$	$85.78_{0.78}$	$58.58_{0.25}$	$62.56_{0.57}$	$56.69_{0.52}$
✓	✓	✓	✓	$\mathbf{64.36}_{0.72}$	$\mathbf{18.55}_{0.04}$	$\mathbf{52.43}_{0.18}$	$\mathbf{86.33}_{0.54}$	$\mathbf{59.06}_{0.48}$	$\mathbf{64.52}_{0.39}$	$\mathbf{57.54}_{0.39}$

4.4 More Analysis

Our method mitigates the adaptation bias, as evidenced by consistent improvements in average omitted-group recall (OGR) and omitted-class accuracy (OCA) across eight datasets. Higher OGR indicates fewer omitted-class samples being misclassified as provided classes, and higher OCA demonstrates better fine-grained recognition of omitted classes. Compared with the best adaptation method for each metric, LSR improves OGR and OCA by **5.06** and **5.72** percentage points at $\gamma = 0.2$, **4.66** and **4.38** percentage points at $\gamma = 0.3$, and **4.22** and **5.08** percentage points at $\gamma = 0.4$, respectively. Moreover, LSR largely preserves the zero-shot CLIP baseline's performance in both metrics and surpasses it in several settings. These results support the effectiveness of reconstructed semantic supervision in reducing adaptation bias toward the provided classes and preserving omitted-class recognition.



(a) $\gamma = 0.2$ (b) $\gamma = 0.3$ (c) $\gamma = 0.4$

Figure 2: Average results of omitted-group recall (%) and omitted-class accuracy (%) across eight datasets at different omission ratios γ.

The reconstructed latent label space contains semantic anchors derived from omitted-class images which captures the majority of omitted-class samples. As shown in Fig. 3, the proportion of anchors derived from omitted-class images increases with the omission ratio. Table 4 further demonstrates that these anchors cover 65.00%~91.66% of omitted classes and capture 64.49%~87.95% of omitted-class samples. Together, these results support the ability of LSR to incorporate omitted-class semantics into the reconstructed latent label space. During prompt tuning, these anchors serve as prediction targets, and $\mathcal{L}_{\mathrm{R}}$ encourages discrimination among the corresponding latent groups. The effectiveness of our design is further supported by its consistent advantage over simple WordNet-based vocabulary expansion, as demonstrated in Appendix G.3.

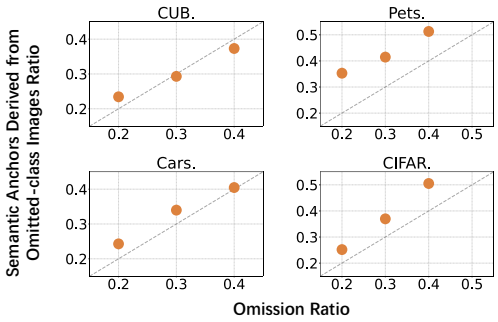


Figure 3: The proportion of semantic anchors derived from omitted-class images at different omission ratios.

Table 4: The omitted classes coverage ratio (%) and omitted-class samples capture ratio (%) of all semantic anchors derived from omitted-class images at different γ.

Metrics	γ	CUB.	Pets.	Cars.	CIFAR.
Omitted classes coverage ratio	0.2	82.50	87.50	87.50	80.00
	0.3	75.00	91.66	83.05	83.33
	0.4	65.00	86.66	65.82	82.50
	Avg.	74.17	88.61	78.79	81.94
Omitted-class samples capture ratio	0.2	86.89	79.84	66.16	64.49
	0.3	81.74	86.90	72.49	71.51
	0.4	86.89	87.95	75.15	82.11
	Avg.	85.17	84.90	71.27	72.70

Evaluation with a larger image encoder. We evaluate LSR with a larger image encoder ViT-L/14 on five datasets. As shown in Table 5, LSR achieves the highest accuracy on all five datasets, exceeding the strongest baseline by an average of **1.89** percentage points. These results demonstrate that our method remains effective with a larger visual backbone.

Evaluation with a stronger foundation model. We further evaluate LSR with PE-Core-B/16 (Bolya et al., 2025) to assess its effectiveness on a stronger foundation model. Table 6 shows that LSR consistently achieves the highest accuracy across all five datasets, reaching an average of 73.96%. Specifically, our method outperforms the strongest baseline on RES and DTD by **3.32** and **2.07** percentage points, respectively. These results illustrate the robustness of our method across different foundation models.

Table 5: Comparison results of all-class accuracy (%) with the image encoder ViT-L/14.

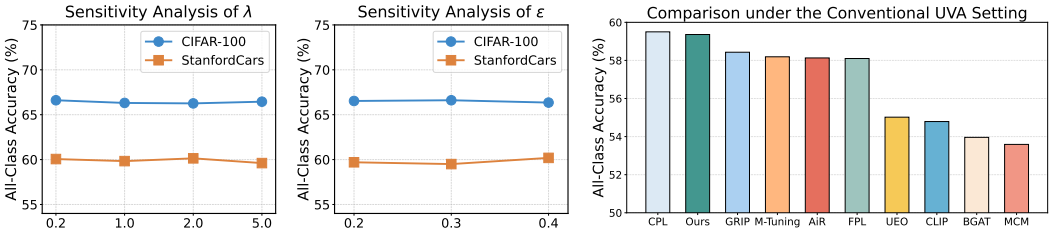
Method	Flw.	FA.	DTD.	RES.	CIFAR.
CLIP	$73.94_{0.00}$	$30.51_{0.00}$	$52.76_{0.00}$	$62.63_{0.00}$	$75.93_{0.00}$
FPL	$76.35_{1.23}$	$30.65_{0.42}$	$52.62_{1.33}$	$68.75_{0.19}$	$79.34_{0.19}$
GRIP	$70.23_{0.60}$	$26.28_{1.53}$	$48.49_{0.67}$	$71.90_{0.86}$	$76.76_{0.49}$
CPL	$65.85_{5.82}$	$28.52_{1.13}$	$49.79_{0.88}$	$70.07_{0.68}$	$75.79_{0.15}$
AiR	$66.09_{1.05}$	$26.07_{1.20}$	$48.90_{3.09}$	$69.05_{0.74}$	$76.69_{0.17}$
MCM	$71.32_{2.65}$	$27.18_{1.11}$	$52.25_{0.94}$	$62.84_{1.21}$	$78.53_{0.32}$
UEO	$75.42_{0.34}$	$27.86_{0.69}$	$53.07_{0.32}$	$68.24_{1.73}$	$78.71_{0.36}$
M-T	$75.11_{1.01}$	$30.16_{1.04}$	$52.15_{0.35}$	$68.64_{0.87}$	$78.87_{0.23}$
BGAT	$71.92_{1.79}$	$29.21_{0.83}$	$52.59_{0.29}$	$60.72_{0.78}$	$78.49_{0.21}$
Ours	$\mathbf{76.68}_{0.64}$	$\mathbf{33.53}_{0.25}$	$\mathbf{54.63}_{0.14}$	$\mathbf{72.49}_{0.69}$	$\mathbf{79.80}_{0.07}$

Table 6: Comparison results of all-class accuracy (%) on the foundation model PE-Core-B/16.

Method	Flw.	FA.	DTD.	RES.	CIFAR.
CLIP	$84.40_{0.00}$	$55.20_{0.00}$	$65.69_{0.00}$	$67.78_{0.00}$	$82.47_{0.00}$
FPL	$85.82_{0.93}$	$54.72_{1.15}$	$66.60_{0.44}$	$70.22_{0.29}$	$82.86_{0.20}$
GRIP	$80.48_{0.71}$	$44.96_{1.24}$	$65.39_{1.46}$	$68.98_{1.08}$	$81.27_{0.60}$
CPL	$83.14_{0.72}$	$44.04_{0.84}$	$64.79_{1.35}$	$68.72_{1.48}$	$81.75_{0.28}$
AiR	$76.84_{2.36}$	$40.79_{0.57}$	$64.88_{1.12}$	$67.00_{1.32}$	$80.87_{0.44}$
MCM	$82.21_{1.10}$	$40.79_{0.95}$	$64.70_{1.24}$	$64.05_{0.64}$	$83.02_{0.02}$
UEO	$82.54_{1.09}$	$46.19_{1.25}$	$66.74_{1.01}$	$71.14_{1.50}$	$83.62_{0.19}$
M-T	$85.40_{0.25}$	$53.84_{0.54}$	$65.80_{0.55}$	$69.25_{0.92}$	$82.94_{0.18}$
BGAT	$79.67_{0.57}$	$50.63_{0.23}$	$63.88_{0.56}$	$65.67_{0.99}$	$82.13_{0.21}$
Ours	$\mathbf{86.37}_{0.39}$	$\mathbf{56.48}_{1.14}$	$\mathbf{68.81}_{0.43}$	$\mathbf{74.46}_{0.92}$	$\mathbf{83.68}_{0.29}$

Sensitivity analysis. We evaluate the sensitivity of LSR to λ (in Eq. 12) and ε (in Appendix F.1) on CIFAR and Cars. As shown in Fig. 4a and Fig. 4b, all-class accuracy remains relatively stable as λ varies from 0.2 to 5.0 and ε varies from 0.2 to 0.4. These results indicate that LSR is robust to the choice of these hyperparameters. Sensitivity analyses on more datasets and parameters are provided in Appendix G.4.

Comparison under the conventional UVA setting. As shown in Fig. 4c, LSR achieves the second-highest all-class accuracy under the conventional UVA setting. The adaptation vocabulary is complete, reducing the benefit of reconstructing a latent label space. Nevertheless, the reconstructed latent label space provides supplementary supervision derived from unlabeled images, which can complement vocabulary-based pseudo-labels and help LSR maintain competitive performance.



(a) (b) (c)

Figure 4: (a) Sensitivity analysis of λ. (b) Sensitivity analysis of ε. (c) Comparison under the conventional UVA setting.

5 Conclusion

In this work, we propose vocabulary-incomplete unsupervised vision-language model adaptation (VI-UVA), a practical yet challenging setting where only a subset of class names is available during adaptation. We identify the adaptation bias induced by vocabulary incompleteness, which leads to vocabulary overfitting and degrades omitted-class recognition. To address this issue, we propose label space reconstruction (LSR), which reconstructs the latent semantic structures from unlabeled data and incorporates them into prompt tuning. By extending supervision beyond the adaptation vocabulary, LSR effectively mitigates vocabulary overfitting and preserves omitted-class recognition. We also derive an omitted-class semantic discrepancy bound for LSR and identify the key factors affecting the bound. Extensive experiments on eight benchmarks show that LSR consistently outperforms existing methods at different omission ratios and remains robust across different visual backbones and foundation models.

AI USE STATEMENT

In this work, we used generative AI tools solely to assist with grammar, spelling, and sentence clarity. We have reviewed all AI-assisted work and bear full responsibility for all content generated by the LLM. The generative AI contributed in no other way to the paper. The research idea, experiment design, and all other content were developed and completed by ourselves.

ETHICS STATEMENT

We confirm that our work on vocabulary-incomplete unsupervised vision-language model adaptation adheres to the ICLR Code of Ethics. The research utilizes publicly available datasets containing no personally identifiable information. As a strictly technical contribution focused on new learning paradigm and algorithmic design, we foresee no ethical issues or potential for misuse.

REPRODUCIBILITY STATEMENT

The complete code for implementing data preprocessing, model training, and evaluation is provided in the Supplementary Material to facilitate the reproduction of our results. Also, our experimental setup is described in Sec. 4.1 and Appendix F.3.

REFERENCES

Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1563–1572, 2016.

Daniel Bolya, Po-Yao Huang, Peize Sun, Jang Hyun Cho, Andrea Madotto, Chen Wei, Tengyu Ma, Jiale Zhi, Jathushan Rajasegaran, Hanoona Bangalath, Junke Wang, Marco Monteiro, Hu Xu, Shiyu Dong, Nikhila Ravi, Shang-Wen Li, Piotr Dollar, and Christoph Feichtenhofer. Perception encoder: The best visual embeddings are not at the output of the network. In *Advances in Neural Information Processing Systems*, 2025.

Yuyan Chen, Nico Lang, B. Christian Schmidt, Aditya Jain, Yves Basset, Sara Beery, Maxim Larrivée, and David Rolnick. Open-insect: Benchmarking open-set recognition of novel species in biodiversity monitoring. *arXiv preprint arXiv:2503.01691*, 2025.

Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883, 2017.

Yung-Sung Chuang, Yang Li, Dong Wang, Ching-Feng Yeh, Kehan Lyu, Ramya Raghavendra, James Glass, Lifei Huang, Jason Weston, Luke Zettlemoyer, Xinlei Chen, Zhuang Liu, Saining Xie, Wen-tau Yih, Shang-Wen Li, and Hu Xu. Meta CLIP 2: A worldwide scaling recipe. In *Advances in Neural Information Processing Systems*, pp. 48009–48036, 2025.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3606–3613, 2014.

Valentin Gabeff, Marc Rußwurm, Devis Tuia, and Alexander Mathis. Wildclip: Scene and animal attribute retrieval from camera trap data with domain-adapted vision-language models. *International Journal of Computer Vision*, 132(9):3770–3786, 2024.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

Tony Huang, Jack Chu, and Fangyun Wei. Unsupervised prompt learning for vision-language models. *arXiv preprint arXiv:2204.03649*, 2022.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*, pp. 554–561, 2013.

Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Gotmare, Shafiq R. Joty, Caiming Xiong, and Steven Chu-Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *Advances in Neural Information Processing Systems*, pp. 9694–9705, 2021.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. BLIP: bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pp. 12888–12900, 2022.

Yongguang Li, Jindong Li, Qi Wang, Qianli Xing, Runliang Niu, Shengsheng Wang, and Menglin Yang. Beyond retraining: Training-free unknown class filtering for source-free open set domain adaptation of vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 23265–23273, 2026.

Jian Liang, Lijun Sheng, Zhengbo Wang, Ran He, and Tieniu Tan. Realistic unsupervised CLIP fine-tuning with universal entropy optimization. In *International Conference on Machine Learning*, pp. 29667–29681, 2024.

Ning Liao, Xiaopeng Zhang, Min Cao, and Junchi Yan. M-tuning: Prompt tuning with mitigated label bias in open-set scenarios. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(6):5885–5899, 2025.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

Cristina Menghini, Andrew Delworth, and Stephen Bach. Enhancing clip with clip: Exploring pseudolabeling for limited-label prompt tuning. In *Advances in Neural Information Processing Systems*, pp. 60984–61007, 2023.

Yifei Ming, Ziyang Cai, Jiuxiang Gu, Yiyou Sun, Wei Li, and Yixuan Li. Delving into out-of-distribution detection with vision-language representations. In *Advances in Neural Information Processing Systems*, pp. 35087–35102, 2022.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP)*, pp. 722–729, 2008.

Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3498–3505, 2012.

Amit Peleg, Naman Deep Singh, and Matthias Hein. Advancing compositional awareness in CLIP with efficient fine-tuning. In *Advances in Neural Information Processing Systems*, pp. 56959–56998, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763, 2021.

Hairui Ren, Fan Tang, He Zhao, Zixuan Wang, Dandan Guo, and Yi Chang. Beyond words: Augmenting discriminative richness via diffusions in unsupervised prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25135–25144, 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, Wei-Lun Chao, and Yu Su. Bioclip: A vision foundation model for the tree of life. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19412–19424, 2024.

Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7492–7501, 2022.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. Technical report, California Institute of Technology, 2011.

Enguang Wang, Zhimao Peng, Zhengyuan Xie, Fei Yang, Xialei Liu, and Ming-Ming Cheng. Get: Unlocking the multi-modal potential of clip for generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20296–20306, 2025a.

Yuchen Wang, Xuefeng Bai, Xiucheng Li, Weili Guan, Liqiang Nie, and Xinyang Chen. Handling imbalanced pseudolabels for vision-language models with concept alignment and confusion-aware calibrated margin. In *International Conference on Machine Learning*, pp. 62357–62372, 2025b.

Size Wu, Wenwei Zhang, Sheng Jin, Wentao Liu, and Chen Change Loy. Aligning bag of regions for open-vocabulary object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15254–15264, 2023.

Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4818–4829, 2024.

Jiahan Zhang, Qi Wei, Feng Liu, and Lei Feng. Candidate pseudolabel learning: Enhancing vision-language models by prompt tuning with unlabeled data. In *International Conference on Machine Learning*, pp. 60004–60020, 2024a.

Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8):5625–5644, 2024b.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16816–16825, 2022a.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022b.

Haowei Zhu, Ye Tian, and Junguo Zhang. Class incremental learning for wildlife biodiversity monitoring in camera trap images. *Ecological Informatics*, 71:101760, 2022.

TABLE OF CONTENTS FOR THE APPENDIX

- A: Related Learning Paradigms 14
- B: Overview and Pseudo-Code of LSR 14
- C: Proof and further analysis of Theorem 1 15
- D: Datasets and Evaluations 18
- E: Comparison Methods 19
- F: Implementation 20
- F.1: Pseudo-Labeled Dataset Generation 20
- F.2: Filtering Procedure of Unreliable Anchors 20
- F.3: Implementation Details 21
- G: More Experiment Results and Analysis 21
- G.1: More Main Results 21
- G.2: More Ablation Studies 21
- G.3: Comparison with WordNet-Based Vocabulary Expansion 21
- G.4: Sensitivity Analysis 22
- H: Compute Resources 23

A Related Learning Paradigms

In Table 7, we compare vocabulary-incomplete unsupervised VLM adaptation (VI-UVA) with open-set recognition (OSR), out-of-distribution detection (OOD), and generalized category discovery (GCD). VI-UVA differs from these related paradigms in terms of both adaptation data and evaluation objectives. During adaptation, VI-UVA requires no labeled samples, whereas OSR and GCD rely on labeled samples from known classes. During evaluation, VI-UVA performs classification over all classes. In contrast, OSR rejects all unknown-class samples as a single unknown category, OOD detection performs binary classification between ID and OOD samples, and GCD assigns known-class samples to known categories while grouping novel-class samples into unnamed clusters.

Table 7: Comparison of VI-UVA with related learning paradigms.

Paradigm	Adaptation data	Evaluation objectives
OSR	Labeled data from known classes.	Classify known samples and reject all unknown samples as a single unknown category.
OOD	Labeled or unlabeled ID data.	Distinguish OOD samples from ID samples without recognizing their semantic categories.
GCD	Labeled data from known classes and unlabeled data from both known and novel classes.	Classify known classes and cluster novel classes.
VI-UVA	Unlabeled data from all classes and an incomplete adaptation vocabulary of class names.	Classify all classes with the complete vocabulary.

B Overview and Pseudo-Code of LSR

Fig. 5 presents the overview of LSR. The pseudo-code of LSR is shown in Algorithm 1.

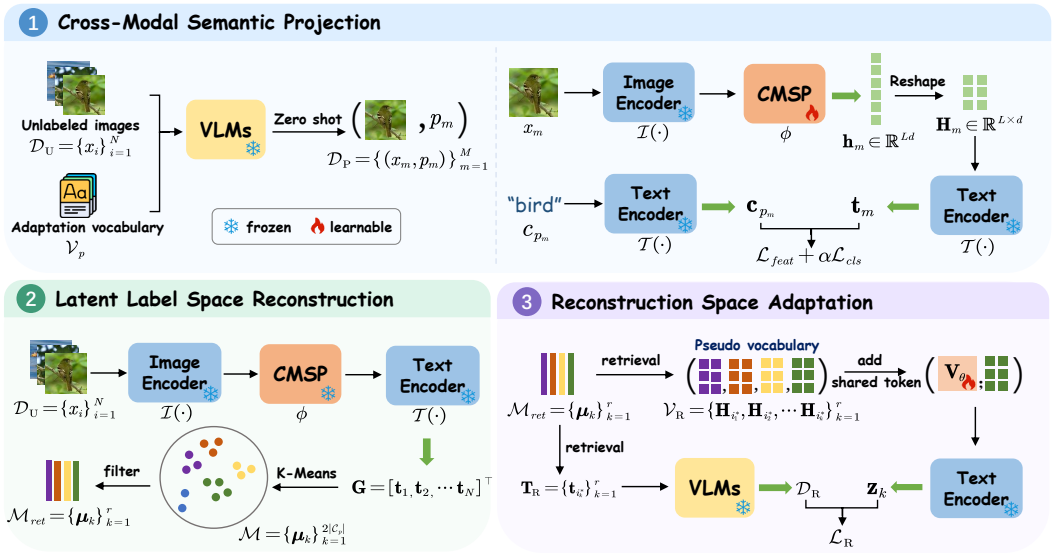


Figure 5: *First*, we train a cross-modal semantic projector (CMSP) using a pseudo-labeled dataset derived from the adaptation vocabulary, mapping visual representations into the textual embedding space. *Second*, we apply the CMSP to all unlabeled images to generate textual embeddings, cluster them into semantic anchors, and filter out unreliable anchors. *Third*, for each retained anchor, we retrieve the corresponding pseudo-token sequence and textual representation to construct a pseudo-vocabulary and a pseudo-labeled dataset, respectively. Then, we perform prompt tuning with the pseudo-vocabulary and pseudo-labeled dataset to preserve omitted-class recognition.

Algorithm 1 Training process of LSR

Input: Unlabeled dataset $\mathcal{D}_{\mathrm{U}}$, adaptation vocabulary $\mathcal{V}_p$, pre-trained image encoder $\mathcal{I}$ and text encoder $\mathcal{T}$, training epochs E_{C} and E_{P}.
Output: Shared learnable context tokens $\mathbf{V}_\theta$.
Freeze the parameters of $\mathcal{I}$ and $\mathcal{T}$.
Stage 1: Train the cross-modal semantic projector.
Extract visual embeddings of $\mathcal{D}_{\mathrm{U}}$ and textual embeddings of $\mathcal{V}_p$.
Construct a balanced pseudo-labeled dataset $\mathcal{D}_{\mathrm{P}}$ with $\mathcal{V}_p$.
Initialize the CMSP parameters ϕ.
for epoch $= 1, \ldots, E_{\mathrm{C}}$ **do**
  for each mini-batch from $\mathcal{D}_{\mathrm{P}}$ **do**
    Compute $\mathcal{L}_{\mathrm{CMSP}} = \mathcal{L}_{\mathrm{feat}} + \alpha\mathcal{L}_{\mathrm{cls}}$.
    Update only ϕ via gradient descent.
  end for
end for
Stage 2: Reconstruct the latent label space.
Freeze the CMSP parameters ϕ.
Generate the pseudo-token sequence $\mathbf{H}_i$ and textual embedding $\mathbf{t}_i$ for each $x_i \in \mathcal{D}_{\mathrm{U}}$.
Perform K-means clustering on $\{\mathbf{t}_i\}_{i=1}^{N}$ into $2|\mathcal{V}_p|$ anchors.
Filter unreliable anchors based on their image support to obtain $\mathcal{M}_{\mathrm{ret}}$.
Stage 3: Optimize the shared context tokens.
for each retained anchor $\boldsymbol{\mu}_k \in \mathcal{M}_{\mathrm{ret}}$ **do**
  Retrieve $i_k^* = \arg\max_{i\in\{1,\ldots,N\}} \mathrm{sim}(\boldsymbol{\mu}_k, \mathbf{t}_i)$.
end for
Construct $\mathcal{V}_{\mathrm{R}} = \{\mathbf{H}_{i_k^*}\}_{k=1}^{r}$ and $\mathbf{T}_{\mathrm{R}} = \{\mathbf{t}_{i_k^*}\}_{k=1}^{r}$.
Construct a balanced pseudo-labeled dataset $\mathcal{D}_{\mathrm{R}}$ with $\mathbf{T}_{\mathrm{R}}$.
Initialize the shared context tokens $\mathbf{V}_\theta$ and keep pseudo-vocabulary $\mathcal{V}_{\mathrm{R}}$ fixed.
for epoch $= 1, \ldots, E_{\mathrm{P}}$ **do**
  for each pair of mini-batches from $\mathcal{D}_{\mathrm{P}}$ and $\mathcal{D}_{\mathrm{R}}$ **do**
    Construct prompt-conditioned textual prototypes using $\mathbf{V}_\theta$ for $\mathcal{V}_p$ and $\mathcal{V}_{\mathrm{R}}$.
    Compute $\mathcal{L}_{\mathrm{all}} = \lambda\mathcal{L}_{\mathrm{R}} + \mathcal{L}_{\mathrm{P}}$.
    Update only $\mathbf{V}_\theta$ via gradient descent.
  end for
end for

C Proof and further analysis of Theorem 1

In this section, we prove Theorem 1. Let $p = |\mathcal{C}_p|$ and $C = |\mathcal{C}_t|$, with provided classes indexed by $\{1, \ldots, p\}$. Throughout the analysis, the textual prototypes are ℓ_2-normalized, i.e., $\|\mathbf{c}_j\|_2 = 1$ for all $j \in \{1, \ldots, C\}$. As in Eq. 2,

$$\mathbf{x}_i = \mathcal{I}(x_i), \quad \mathbf{t}_i = \mathcal{F}(\mathbf{x}_i), \quad \mathcal{F} = \mathcal{T} \circ \mathcal{R}_{L\times d} \circ f_\phi. \tag{14}$$

Let $\mathcal{I}_o$ index the omitted-class samples in $\mathcal{D}_U$ and $N_o = |\mathcal{I}_o|$. For $i \in \mathcal{I}_o$, denote $\mathbf{c}_{y_i}$ as the textual prototype of its true class. Unsubscripted norms are Euclidean.

Assumption 1. *Suppose that $\mathcal{F}$ is ℓ-Lipschitz and its Jacobian is κ-Lipschitz on a convex domain Ω containing the relevant visual representations. For finite $\ell, \kappa \geq 0$ and all $\mathbf{u}, \mathbf{v} \in \Omega$,*

$$\begin{aligned} \|\mathcal{F}(\mathbf{u}) - \mathcal{F}(\mathbf{v})\| &\leq \ell\|\mathbf{u} - \mathbf{v}\|, \\ \|D\mathcal{F}(\mathbf{u}) - D\mathcal{F}(\mathbf{v})\|_{\mathrm{op}} &\leq \kappa\|\mathbf{u} - \mathbf{v}\|, \end{aligned} \tag{15}$$

where $\|\cdot\|_{\mathrm{op}}$ is the induced operator norm.

Define the fixed semantic response over $\mathcal{C}_t$ by

$$q_j(\mathbf{a}) = \frac{\exp(\mathbf{a}^\top \mathbf{c}_j)}{\sum_{k=1}^{C} \exp(\mathbf{a}^\top \mathbf{c}_k)}, \quad q(\mathbf{a}) = (q_j(\mathbf{a}))_{j=1}^{C}. \tag{16}$$

Keep $\mathbf{T}_R = \{\mathbf{t}_{i_k^*}\}_{k=1}^r$ as constructed by the cosine retrieval in Sec. 3.3. For each $i \in \mathcal{I}_o$, define

$$\widehat{\mathbf{t}}_i \in \arg\min_{\mathbf{a}\in\mathbf{T}_R} \|\mathbf{t}_i - \mathbf{a}\|, \quad d_i = \|\mathbf{t}_i - \widehat{\mathbf{t}}_i\|, \quad \mathfrak{r}_o = \left(\frac{1}{N_o}\sum_{i\in\mathcal{I}_o} d_i^2\right)^{1/2}. \tag{17}$$

The discrepancy in Theorem 1 is defined as a KL divergence:

$$\mathcal{E}_o^{\mathrm{rec}} = \frac{1}{N_o}\sum_{i\in\mathcal{I}_o} D_{\mathrm{KL}}\Big(q(\mathbf{c}_{y_i}) \,\Big\|\, q(\widehat{\mathbf{t}}_i)\Big). \tag{18}$$

Lemma 1. *Under Assumption 1, for any* $\mathbf{u}, \mathbf{v} \in \Omega$,

$$\|\mathcal{F}(\mathbf{u}) - \mathcal{F}(\mathbf{v}) - D\mathcal{F}(\mathbf{v})(\mathbf{u}-\mathbf{v})\| \le \frac{\kappa}{2}\|\mathbf{u}-\mathbf{v}\|^2. \tag{19}$$

Consequently, if $a_s \ge 0$, $\sum_s a_s = 1$, *and* $\overline{\mathbf{u}} = \sum_s a_s \mathbf{u}_s$ *with* $\mathbf{u}_s \in \Omega$, *then*

$$\left\|\mathcal{F}(\overline{\mathbf{u}}) - \sum_s a_s \mathcal{F}(\mathbf{u}_s)\right\| \le \frac{\kappa}{2}\sum_s a_s \|\mathbf{u}_s - \overline{\mathbf{u}}\|^2. \tag{20}$$

Proof. Set $\mathbf{h} = \mathbf{u} - \mathbf{v}$. Convexity keeps the segment $\mathbf{v} + t\mathbf{h}$ inside Ω. The fundamental theorem of calculus gives

$$\mathcal{F}(\mathbf{u}) - \mathcal{F}(\mathbf{v}) - D\mathcal{F}(\mathbf{v})\mathbf{h} = \int_0^1 \big[D\mathcal{F}(\mathbf{v}+t\mathbf{h}) - D\mathcal{F}(\mathbf{v})\big]\mathbf{h}\, dt. \tag{21}$$

Its norm is at most $\int_0^1 \kappa t\|\mathbf{h}\|^2 dt = \kappa\|\mathbf{h}\|^2/2$, proving Eq. 19. Apply this inequality to each pair $(\mathbf{u}_s, \overline{\mathbf{u}})$ and take the weighted sum. The linear terms cancel because $\sum_s a_s(\mathbf{u}_s - \overline{\mathbf{u}}) = 0$. The triangle inequality gives Eq. 20. □

For $\mathcal{D}_P = \{(x_m, p_m)\}_{m=1}^M$, define $\mathcal{P}_j = \{m : p_m = j\}$, $n_j = |\mathcal{P}_j|$, and

$$\begin{aligned}
\overline{\mathbf{x}}_j &= \frac{1}{n_j}\sum_{m\in\mathcal{P}_j} \mathbf{x}_m, \\
e_j &= \frac{1}{n_j}\sum_{m\in\mathcal{P}_j} \|\mathbf{t}_m - \mathbf{c}_j\|_1, \\
v_j &= \frac{1}{n_j}\sum_{m\in\mathcal{P}_j} \|\mathbf{x}_m - \overline{\mathbf{x}}_j\|^2, \\
\varepsilon_{\mathrm{proj}} &= \max_{1\le j\le p}\left(e_j + \frac{\kappa}{2}v_j\right).
\end{aligned} \tag{22}$$

Lemma 2. *Under Assumption 1, for every* $j \in \{1, \dots, p\}$,

$$\|\mathcal{F}(\overline{\mathbf{x}}_j) - \mathbf{c}_j\| \le e_j + \frac{\kappa}{2}v_j \le \varepsilon_{\mathrm{proj}}. \tag{23}$$

Proof. Apply Lemma 1 to the representations in $\mathcal{P}_j$ with weights $1/n_j$. Inserting their mean generated representation gives

$$\begin{aligned}
\|\mathcal{F}(\overline{\mathbf{x}}_j) - \mathbf{c}_j\| &\le \left\|\mathcal{F}(\overline{\mathbf{x}}_j) - \frac{1}{n_j}\sum_{m\in\mathcal{P}_j}\mathbf{t}_m\right\| + \left\|\frac{1}{n_j}\sum_{m\in\mathcal{P}_j}(\mathbf{t}_m - \mathbf{c}_j)\right\| \\
&\le \frac{\kappa}{2}v_j + \frac{1}{n_j}\sum_{m\in\mathcal{P}_j}\|\mathbf{t}_m - \mathbf{c}_j\|_1 \\
&= \frac{\kappa}{2}v_j + e_j,
\end{aligned} \tag{24}$$

where $\|\cdot\|_2 \le \|\cdot\|_1$ gives the second line. Taking the maximum over groups proves the lemma. □

Let $\Delta_p = \{\mathbf{w} \in \mathbb{R}^p : w_j \geq 0, \sum_{j=1}^p w_j = 1\}$. For the same $\mathbf{w} \in \Delta_p$, set

$$\mathbf{x}_\mathbf{w} = \sum_{j=1}^p w_j \overline{\mathbf{x}}_j, \qquad \mathbf{c}_\mathbf{w} = \sum_{j=1}^p w_j \mathbf{c}_j.$$

Define the shared-recomposition discrepancy by

$$\begin{aligned} b_i &= \min_{\mathbf{w}\in\Delta_p} \left\{ \ell\|\mathbf{x}_i - \mathbf{x}_\mathbf{w}\| + \|\mathbf{c}_{y_i} - \mathbf{c}_\mathbf{w}\| + \frac{\kappa}{2} \sum_{j=1}^p w_j \|\overline{\mathbf{x}}_j - \mathbf{x}_\mathbf{w}\|^2 \right\}, \\ \mathfrak{b}_o &= \left(\frac{1}{N_o} \sum_{i\in\mathcal{I}_o} b_i^2 \right)^{1/2}. \end{aligned} \tag{25}$$

The three terms of b_i are the visual residual, textual residual, and nonlinear recomposition remainder. They are evaluated with common weights $\mathbf{w}$, which may vary across samples. The minimum exists by continuity on the compact simplex.

Lemma 3. *Under Assumption 1, every $i \in \mathcal{I}_o$ satisfies*

$$\|\mathbf{t}_i - \mathbf{c}_{y_i}\| \leq \varepsilon_{\text{proj}} + b_i. \tag{26}$$

Proof. For any $\mathbf{w} \in \Delta_p$, apply the triangle inequality and insert $\mathcal{F}(\mathbf{x}_\mathbf{w})$, $\sum_j w_j \mathcal{F}(\overline{\mathbf{x}}_j)$, and $\mathbf{c}_\mathbf{w}$ to obtain

$$\begin{aligned} \|\mathbf{t}_i - \mathbf{c}_{y_i}\| &\leq \|\mathcal{F}(\mathbf{x}_i) - \mathcal{F}(\mathbf{x}_\mathbf{w})\| \\ &\quad + \left\| \mathcal{F}(\mathbf{x}_\mathbf{w}) - \sum_j w_j \mathcal{F}(\overline{\mathbf{x}}_j) \right\| \\ &\quad + \left\| \sum_j w_j \big(\mathcal{F}(\overline{\mathbf{x}}_j) - \mathbf{c}_j \big) \right\| + \|\mathbf{c}_\mathbf{w} - \mathbf{c}_{y_i}\|. \end{aligned} \tag{27}$$

Assumption 1 bounds the first term by $\ell\|\mathbf{x}_i - \mathbf{x}_\mathbf{w}\|$, Lemma 1 bounds the second by $\frac{\kappa}{2}\sum_j w_j \|\overline{\mathbf{x}}_j - \mathbf{x}_\mathbf{w}\|^2$, and Lemma 2 bounds the third by $\sum_j w_j \varepsilon_{\text{proj}} = \varepsilon_{\text{proj}}$ which proves the lemma. □

Lemma 4. *For $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$,*

$$D_{\text{KL}}(q(\mathbf{a}) \,\|\, q(\mathbf{b})) \leq \frac{1}{2}\|\mathbf{a} - \mathbf{b}\|^2. \tag{28}$$

Proof. Let $\Psi(\mathbf{a}) = \log \sum_{j=1}^C \exp(\mathbf{a}^\top \mathbf{c}_j)$. Differentiation gives

$$\nabla\Psi(\mathbf{a}) = \sum_j q_j(\mathbf{a}) \mathbf{c}_j, \quad \nabla^2\Psi(\mathbf{a}) = \sum_j q_j(\mathbf{a}) \mathbf{c}_j \mathbf{c}_j^\top - \nabla\Psi(\mathbf{a}) \nabla\Psi(\mathbf{a})^\top. \tag{29}$$

The Hessian is a covariance matrix. Hence, for any $\mathbf{h}$,

$$0 \leq \mathbf{h}^\top \nabla^2\Psi(\mathbf{a}) \mathbf{h} \leq \sum_j q_j(\mathbf{a}) (\mathbf{h}^\top \mathbf{c}_j)^2 \leq \sum_j q_j(\mathbf{a}) \|\mathbf{h}\|^2 \|\mathbf{c}_j\|^2 = \|\mathbf{h}\|^2, \tag{30}$$

where the last step follows from the ℓ_2-normalization of the textual prototypes, i.e., $\|\mathbf{c}_j\| = 1$ for all j. Since $\log q_j(\mathbf{a}) = \mathbf{a}^\top \mathbf{c}_j - \Psi(\mathbf{a})$, the KL divergence equals

$$\begin{aligned} D_{\text{KL}}(q(\mathbf{a}) \,\|\, q(\mathbf{b})) &= \sum_j q_j(\mathbf{a}) \left[(\mathbf{a} - \mathbf{b})^\top \mathbf{c}_j - \Psi(\mathbf{a}) + \Psi(\mathbf{b}) \right] \\ &= \Psi(\mathbf{b}) - \Psi(\mathbf{a}) - \nabla\Psi(\mathbf{a})^\top (\mathbf{b} - \mathbf{a}). \end{aligned} \tag{31}$$

Set $\mathbf{h} = \mathbf{b} - \mathbf{a}$. Taylor's integral remainder and Eq. 30 give

$$\begin{aligned} D_{\text{KL}}(q(\mathbf{a}) \,\|\, q(\mathbf{b})) &= \int_0^1 (1-t) \mathbf{h}^\top \nabla^2\Psi(\mathbf{a} + t\mathbf{h}) \mathbf{h} \, dt \\ &\leq \|\mathbf{h}\|^2 \int_0^1 (1-t) dt = \frac{1}{2}\|\mathbf{h}\|^2. \end{aligned} \tag{32}$$

□

Proof of Theorem 1. Under Assumption 1, Lemma 3 and Eq. 17 imply

$$\|\widehat{\mathbf{t}}_i - \mathbf{c}_{y_i}\| \le \|\widehat{\mathbf{t}}_i - \mathbf{t}_i\| + \|\mathbf{t}_i - \mathbf{c}_{y_i}\| \le d_i + \varepsilon_{\text{proj}} + b_i. \tag{33}$$

Applying Lemma 4 to $(\mathbf{c}_{y_i}, \widehat{\mathbf{t}}_i)$ and averaging yields

$$\mathcal{E}_o^{\text{rec}} \le \frac{1}{2N_o} \sum_{i \in \mathcal{I}_o} \|\widehat{\mathbf{t}}_i - \mathbf{c}_{y_i}\|^2 \le \frac{1}{2N_o} \sum_{i \in \mathcal{I}_o} (\varepsilon_{\text{proj}} + b_i + d_i)^2. \tag{34}$$

The Euclidean triangle inequality gives

$$\begin{aligned} \left[\frac{1}{N_o} \sum_{i \in \mathcal{I}_o} (\varepsilon_{\text{proj}} + b_i + d_i)^2\right]^{1/2} &\le \varepsilon_{\text{proj}} + \left(\frac{1}{N_o} \sum_{i \in \mathcal{I}_o} b_i^2\right)^{1/2} + \left(\frac{1}{N_o} \sum_{i \in \mathcal{I}_o} d_i^2\right)^{1/2} \\ &= \varepsilon_{\text{proj}} + \mathfrak{b}_o + \mathfrak{r}_o. \end{aligned} \tag{35}$$

Squaring Eq. 35 and substituting into Eq. 34 establishes

$$\mathcal{E}_o^{\text{rec}} \le \frac{1}{2} \left(\varepsilon_{\text{proj}} + \mathfrak{b}_o + \mathfrak{r}_o\right)^2. \tag{36}$$

□

Analysis of Theorem 1. $\varepsilon_{\text{proj}}$ is determined by the quality of cross-modal semantic projection on the provided classes. It becomes smaller when the generated textual representations are better aligned with their corresponding textual prototypes and the pseudo-labeled groups exhibit more compact visual structures. In LSR, the feature regression objective $\mathcal{L}_{\text{feat}}$ in Eq. 3 directly encourages such semantic alignment, while balanced high-confidence sample selection helps construct more compact provided-class groups.

$\mathfrak{b}_o$ characterizes the structural consistency between the visual and textual spaces which depends on the dataset. It is small when an omitted-class sample and its textual semantics can be well recomposed from the provided-class structures using consistent convex weights. Therefore, a smooth cross-modal mapping that preserves the relative structure between the two modalities facilitates semantic transfer from provided to omitted classes.

Finally, $\mathfrak{r}_o$ depends on how well the retrieval textual representations cover the generated textual space. In LSR, over-clustering provides additional semantic anchors, filtering removes unreliable anchors, and representative retrieval in Eq. 6 associates the retained anchors with actual generated textual representations. These operations jointly improve the coverage and reliability of the retrieval textual representations, thereby reducing its approximation error.

Overall, accurate projection, lower cross-modal recomposition discrepancy and representation approximation error jointly lead to a tighter bound.

D Datasets and Evaluations

We conduct experiments on eight image classification datasets from distinct domains, including DTD (DTD.) (Cimpoi et al., 2014), FGVC-Aircraft (FA.) (Maji et al., 2013), CUB (CUB.) (Wah et al., 2011), Flowers102 (Flw.) (Nilsback & Zisserman, 2008), OxfordPets (Pets.) (Parkhi et al., 2012), StanfordCars (Cars.) (Krause et al., 2013), RESISC45 (RES.) (Cheng et al., 2017), and CIFAR100 (CIFAR.) (Krizhevsky, 2009). Dataset statistics are presented in Table 8. All datasets adopt the default training-test splits.

In our experiments, we evaluate methods with three metrics, including omitted-class accuracy, omitted-group recall, and all-class accuracy. Let $\mathcal{D}_{\text{test}} = \{(x_i, y_i)\}_{i=1}^N$ denote the test set, where predictions are made over the set of all class names $\mathcal{C}_t = \mathcal{C}_p \cup \mathcal{C}_o$. Let $\hat{y}_i$ be the predicted label of x_i, and define $\mathcal{I}_o = \{i \mid y_i \in \mathcal{C}_o\}$, $N_o = |\mathcal{I}_o|$.

Omitted-class accuracy (OCA) measures the classification accuracy of omitted-class samples:

$$\text{OCA} = \frac{1}{N_o} \sum_{i \in \mathcal{I}_o} \mathbf{1}(\hat{y}_i = y_i) \times 100\%. \tag{37}$$

Table 8: Dataset statistics.

	DTD.	FA.	CUB.	Flw.	Pets.	Cars.	RES.	CIFAR.
Class number	47	100	200	102	37	196	45	100
Training set size	3760	6667	5594	2040	3680	8144	6300	50000
Testing set size	1880	3333	5794	6149	3669	8041	25200	10000
Balanced	✓	✓	✗	✗	✗	✗	✓	✓

Omitted-group recall (OGR) treats all omitted classes as a single group and measures the proportion of omitted-class samples that are predicted as any omitted class:

$$\mathrm{OGR} = \frac{1}{N_o} \sum_{i \in \mathcal{I}_o} \mathbf{1}(\hat{y}_i \in \mathcal{C}_o) \times 100\%. \tag{38}$$

Unlike OCA, OGR does not require the exact omitted-class label to be correctly predicted. Instead, it reflects how well the model avoids misclassifying omitted-class samples as provided classes.

All-class accuracy (ACC) measures the overall classification accuracy across both provided and omitted classes:

$$\mathrm{ACC} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}(\hat{y}_i = y_i) \times 100\%. \tag{39}$$

$\mathbf{1}(\cdot)$ denotes the indicator function, which equals 1 when the condition is satisfied and 0 otherwise.

E Comparison Methods

We compare our method with ten SOTA methods, including CLIP (Radford et al., 2021), Few Pseudo-Labels (FPL) (Menghini et al., 2023), Grow and Refine Iteratively Pseudo-labels (GRIP) (Menghini et al., 2023), Candidate Pseudolabel Learning (CPL) (Zhang et al., 2024a), Universal Entropy Optimization (UEO) (Liang et al., 2024), Augmenting Discriminative Richness (AiR) (Ren et al., 2025), Concept-Adaptive Pseudo-labeling (CAP) (Wang et al., 2025b), Maximum Concept Matching (MCM) (Ming et al., 2022), M-Tuning (Liao et al., 2025), and BGAT (Li et al., 2026). Among these methods, MCM, M-Tuning, and BGAT were originally designed for open-set scenarios, and we adapt them to the VI-UVA setting. The details of each method are presented as follows:

- **CLIP:** CLIP exhibits zero-shot capability for classification.
- **FPL:** FPL filters the top-k high-confidence pseudo-labeled samples for each class to construct a balanced pseudo-label dataset for model training.
- **GRIP:** GRIP predefines the number of iterations. Within each iteration, it regenerates pseudo-labels and filters the top-k high-confidence pseudo-labeled samples for each class to construct a balanced pseudo-label dataset, where k increases linearly as iterations proceed.
- **CPL:** CPL uses intra- and inter-instance selected candidate pseudo-labels instead of hard pseudo-labels, effectively reducing label noise and class imbalance for better adaptation.
- **UEO:** UEO is a universal entropy optimization method for realistic unsupervised CLIP fine-tuning. It uses sample confidence to minimize entropy for in-distribution samples and maximize marginal entropy for OOD samples.
- **AiR:** AiR utilizes LoRA-fine-tuned diffusion models to generate class-representative synthetic images and builds an auxiliary classifier for high-quality pseudo-labels.
- **CAP:** CAP proposes concept alignment to fix semantic-visual misalignment, and a confusion-aware calibrated margin to improve discrimination between similar classes. It also adopts dual independent adapters to separately learn balanced pseudo-labels and high-confidence unlabeled samples.
- **MCM:** MCM is adapted to VI-UVA by computing concept-matching scores over the adaptation vocabulary and using these scores to select samples for prompt tuning.

- **M-Tuning:** M-Tuning is adapted to VI-UVA by augmenting the adaptation vocabulary with auxiliary open words and incorporating them into prompt tuning. These additional words expand the set of textual concepts considered during optimization, helping mitigate the model's bias toward the provided classes.
- **BGAT:** We adapt BGAT to VI-UVA by applying a Box–Cox transformation and a two-component Gaussian mixture model to sample scores computed over the adaptation vocabulary, yielding a data-adaptive threshold. This threshold filters out potential omitted-class samples.

F IMPLEMENTATION

F.1 PSEUDO-LABELED DATASET GENERATION

We first exploit the zero-shot predictions of the pre-trained VLM to assign each x_i in $\mathcal{D}_{\mathrm{U}}$ to the most likely provided class. Let $\hat{y}_i \in \mathcal{C}_p$ denote the predicted class of x_i. For each provided class $c \in \mathcal{C}_p$, we define the set of assigned sample indices and its size as

$$\mathcal{S}_c = \{i \mid \hat{y}_i = c\}, \;\; n_c = |\mathcal{S}_c|. \tag{40}$$

Then, the provided classes are sorted in ascending order of their sample counts. Let π denote the resulting ordering, such that

$$n_{\pi(1)} \leq n_{\pi(2)} \leq \cdots \leq n_{\pi(|\mathcal{C}_p|)}. \tag{41}$$

The first $\lfloor \varepsilon \, |\mathcal{C}_p| \rfloor$ classes constitute the tail-class subset $\mathcal{C}_{\mathrm{tail}} = \{\pi(1), \ldots, \pi(|\mathcal{C}_{\mathrm{tail}}|)\}$, where $\varepsilon \in (0, 1)$ denotes the ratio of tail classes. To mitigate class imbalance, we use their average sample count to determine a per-class upper bound:

$$U = \left\lfloor \frac{1}{|\mathcal{C}_{\mathrm{tail}}|} \sum_{j=1}^{|\mathcal{C}_{\mathrm{tail}}|} n_{\pi(j)} \right\rfloor. \tag{42}$$

For each provided class, we compute the confidence score of each x_i in $\mathcal{D}_{\mathrm{U}}$ for that class and select the U highest-confidence images. Each selected image is paired with the corresponding class as its pseudo-label, and these pairs are combined into a balanced pseudo-labeled dataset $\mathcal{D}_{\mathrm{P}}$ for training the cross-modal semantic projector. To construct $\mathcal{D}_{\mathrm{R}}$, we follow the same procedure, using $\mathcal{V}_R$ in place of $\mathcal{C}_p$.

F.2 FILTERING PROCEDURE OF UNRELIABLE ANCHORS

Over-clustering may produce unreliable semantic anchors that capture outliers or noise. We evaluate the reliability of each candidate anchor by the number of unlabeled images assigned to it. Since the semantic anchors lie in the textual embedding space shared with the visual embedding space, we calculate the cosine similarity between each visual embedding $\mathbf{x}_i$ and every semantic anchor $\boldsymbol{\mu}_k$ and assign $\mathbf{x}_i$ to its most similar anchor:

$$a_i = \underset{k \in \{1, \ldots, K\}}{\arg\max} \frac{\boldsymbol{\mu}_k^\top \mathbf{x}_i}{\|\boldsymbol{\mu}_k\|_2 \, \|\mathbf{x}_i\|_2}. \tag{43}$$

where K is the number of candidate anchors. The support count of the anchor $\boldsymbol{\mu}_k$ is defined as:

$$S_k = \sum_{i=1}^{N} \mathbf{1}(a_i = k), \tag{44}$$

where $\mathbf{1}(\cdot)$ is the indicator function, which equals 1 when the specified condition is satisfied and 0 otherwise. N represents the number of images in $\mathcal{D}_{\mathrm{U}}$. We retain anchors whose support counts exceed the threshold $\lfloor \delta N / \, |\mathcal{C}_p| \rfloor$, where δ controls the minimum support count and is fixed at 0.05 in our implementation. This procedure filters out anchors supported by very few images, helping to suppress noise while retaining the dominant semantic structures in the unlabeled data.

F.3 Implementation Details

We adopt CLIP with a ViT-B/32 image encoder and set the number of learnable context tokens S to 16. The parameter ε for constructing balanced pseudo-labeled datasets takes values in $\{0.2, 0.3, 0.4\}$. The batch size is set to 32. We train CMSP for 50 epochs using AdamW with a learning rate of 10^{-4} and a weight decay of 10^{-4}. The length of the pseudo-token sequence L is set to 7, and the loss coefficient α is fixed at 0.1. The parameter δ for filtering unreliable anchors is fixed at 0.05. For textual prompt tuning, we optimize the shared context tokens using SGD for a maximum of 50 epochs, with an initial learning rate of 0.01, a cosine annealing schedule, and a weight decay of 0.05. The temperature parameter τ is fixed at 1.0, and the loss coefficient λ takes values in $\{0.2, 1.0, 2.0, 5.0\}$. Sensitivity analyses of the parameters are presented in Sec 4.4 and Appendix G.4.

G More Experiment Results and Analysis

G.1 More Main Results

Table 9 compares the all-class accuracy of different methods at omission ratio of $\gamma = 0.4$. All adaptation methods except LSR perform worse than the CLIP baseline on average, while LSR outperforms the CLIP baseline by an average of **2.34** percentage points. These results demonstrate the effectiveness of LSR at a larger omission ratio.

Table 9: Comparison results of all-class accuracy (%) at $\gamma = 0.4$. Values are expressed as mean_{std}. The highest accuracies are bold.

Methods	Flw.	FA.	CUB.	DTD.	Pets.	Cars.	RES.	CIFAR.	Avg.
CLIP	$63.63_{0.00}$	$17.64_{0.00}$	$51.77_{0.00}$	$43.19_{0.00}$	$84.51_{0.00}$	$58.93_{0.00}$	$54.48_{0.00}$	$64.19_{0.00}$	$54.79_{0.00}$
FPL	$62.49_{0.24}$	$15.61_{0.94}$	$49.41_{0.38}$	$43.99_{0.46}$	$88.01_{1.22}$	$53.83_{0.47}$	$58.28_{0.74}$	$64.93_{0.68}$	$54.57_{0.64}$
GRIP	$55.36_{0.53}$	$12.67_{1.08}$	$39.51_{0.86}$	$42.25_{1.58}$	$75.39_{2.98}$	$45.02_{0.97}$	$57.35_{2.30}$	$61.31_{0.35}$	$48.61_{1.33}$
CPL	$54.74_{0.94}$	$14.20_{0.35}$	$44.24_{0.97}$	$41.12_{1.99}$	$79.54_{1.02}$	$43.81_{0.46}$	$55.98_{0.71}$	$58.54_{0.88}$	$49.02_{0.92}$
CAP	$52.78_{3.44}$	$13.64_{1.67}$	$44.66_{0.91}$	$40.66_{1.58}$	$73.21_{0.19}$	$50.27_{0.32}$	$60.94_{0.58}$	$61.70_{0.04}$	$49.73_{1.09}$
AiR	$55.69_{1.88}$	$12.41_{0.65}$	$37.94_{0.12}$	$42.06_{1.18}$	$76.01_{2.08}$	$45.14_{0.50}$	$56.09_{0.76}$	$61.09_{0.50}$	$48.30_{0.96}$
MCM	$59.32_{0.37}$	$14.29_{2.09}$	$44.76_{0.73}$	$39.49_{1.22}$	$80.20_{2.98}$	$50.11_{1.21}$	$49.73_{0.35}$	$63.55_{0.38}$	$50.18_{1.17}$
UEO	$58.35_{1.48}$	$8.22_{1.89}$	$42.57_{0.77}$	$40.43_{0.87}$	$80.94_{1.03}$	$49.04_{2.02}$	$57.38_{0.59}$	$45.56_{2.58}$	$47.81_{1.40}$
M-Tuning	$61.55_{0.28}$	$15.98_{0.96}$	$49.36_{0.33}$	$43.01_{0.27}$	$\mathbf{88.01}_{0.31}$	$53.39_{0.77}$	$58.31_{1.60}$	$64.48_{0.40}$	$54.26_{0.62}$
BGAT	$58.06_{0.96}$	$16.69_{0.27}$	$46.57_{0.66}$	$39.18_{0.76}$	$82.80_{0.50}$	$55.15_{0.72}$	$51.95_{0.20}$	$64.54_{0.13}$	$51.87_{0.53}$
LSR (Ours)	$\mathbf{64.36}_{0.73}$	$\mathbf{18.55}_{0.05}$	$\mathbf{52.43}_{0.18}$	$\mathbf{45.59}_{0.52}$	$86.33_{0.55}$	$\mathbf{59.06}_{0.48}$	$\mathbf{64.52}_{0.39}$	$\mathbf{66.18}_{0.25}$	$\mathbf{57.13}_{0.39}$

G.2 More Ablation Studies

As shown in Table 10, incorporating $\mathcal{L}_{\text{R}}$ improves omitted-class accuracy across all six datasets, with an average gain of **3.39** percentage points. By introducing supervision from the reconstructed latent label space, $\mathcal{L}_{\text{R}}$ encourages prompts to retain semantic distinctions beyond the adaptation vocabulary.

Table 10: Comparison of omitted-class accuracy (%) for ablation variants w and w/o $\mathcal{L}_{\text{R}}$ at $\gamma = 0.4$. Values are expressed as mean_{std}. The highest accuracies are bold.

$\mathcal{L}_{\text{R}}$	Flw.	FA.	CUB.	Pets.	Cars.	RES.	Avg.
✗	$66.92_{1.14}$	$18.30_{1.75}$	$39.18_{2.11}$	$85.10_{1.31}$	$57.81_{0.49}$	$57.96_{0.00}$	$54.21_{1.25}$
✓	$\mathbf{72.00}_{0.69}$	$\mathbf{20.33}_{0.12}$	$\mathbf{44.16}_{0.24}$	$\mathbf{89.31}_{0.47}$	$\mathbf{59.54}_{0.30}$	$\mathbf{60.29}_{0.41}$	$\mathbf{57.60}_{0.37}$

G.3 Comparison with WordNet-Based Vocabulary Expansion

To investigate an alternative way to mitigate vocabulary overfitting, we expand the adaptation vocabulary with additional words retrieved from WordNet. For each unlabeled image, we seek words that better match its visual embedding than the class names in the adaptation vocabulary. These words supplement the adaptation vocabulary to provide supervision beyond the provided classes.

Table 11 shows that LSR consistently outperforms this strategy across all eight datasets and three omission ratios. At $\gamma = 0.2$, 0.3, and 0.4, LSR yields gains of **4.08**, **4.53**, and **4.68** percentage points in all-class accuracy. These results suggest that retrieving semantically similar words alone is insufficient to construct effective supervision for adaptation. Although additional words may better match individual images, higher image-text similarity does not necessarily imply that they capture the distinctions among classes. Retrieved words may describe shared attributes or broad concepts, providing limited class-discriminative supervision. In contrast, LSR constructs semantic anchors by clustering representations from the unlabeled data, allowing the reconstructed label space to reflect its latent semantic structures.

Table 11: Comparison of all-class accuracy (%) between WordNet and LSR at different γ.Values are expressed as mean_{std}. The highest accuracies are bold.

γ	Methods	Flw.	FA.	CUB.	DTD.	Pets.	Cars.	RES.	CIFAR.	Avg.
0.2	WordNet	$62.77_{0.42}$	$16.67_{0.23}$	$51.04_{0.73}$	$42.07_{0.54}$	$85.79_{2.07}$	$57.22_{0.53}$	$57.12_{1.13}$	$62.86_{0.45}$	$54.44_{0.76}$
	LSR (Ours)	$\mathbf{66.89}_{0.60}$	$\mathbf{19.34}_{0.41}$	$\mathbf{53.27}_{0.22}$	$\mathbf{47.02}_{0.54}$	$\mathbf{88.49}_{0.60}$	$\mathbf{60.15}_{0.34}$	$\mathbf{65.82}_{0.35}$	$\mathbf{67.14}_{0.53}$	$\mathbf{58.52}_{0.45}$
0.3	WordNet	$55.34_{0.76}$	$16.61_{0.20}$	$50.56_{0.24}$	$40.34_{0.87}$	$86.85_{0.69}$	$55.94_{0.73}$	$56.79_{0.73}$	$63.08_{0.36}$	$53.19_{0.57}$
	LSR (Ours)	$\mathbf{65.50}_{0.42}$	$\mathbf{18.92}_{0.18}$	$\mathbf{53.00}_{0.50}$	$\mathbf{45.32}_{0.52}$	$\mathbf{87.53}_{0.33}$	$\mathbf{59.68}_{0.29}$	$\mathbf{65.26}_{0.41}$	$\mathbf{66.53}_{0.14}$	$\mathbf{57.72}_{0.35}$
0.4	WordNet	$56.64_{0.39}$	$15.31_{0.45}$	$49.18_{1.53}$	$41.01_{0.28}$	$84.11_{1.23}$	$54.29_{0.78}$	$56.31_{0.32}$	$62.71_{1.36}$	$52.45_{0.79}$
	LSR (Ours)	$\mathbf{64.36}_{0.73}$	$\mathbf{18.55}_{0.05}$	$\mathbf{52.43}_{0.18}$	$\mathbf{45.59}_{0.52}$	$\mathbf{86.33}_{0.55}$	$\mathbf{59.06}_{0.48}$	$\mathbf{64.52}_{0.39}$	$\mathbf{66.18}_{0.25}$	$\mathbf{57.13}_{0.39}$

G.4 Sensitivity Analysis

We extend the sensitivity analysis of λ and ε to additional datasets. As shown in Fig. 6, LSR maintains relatively stable all-class accuracy as either parameter varies within the tested range. These results further support the robustness of LSR to the choice of these two hyperparameters.

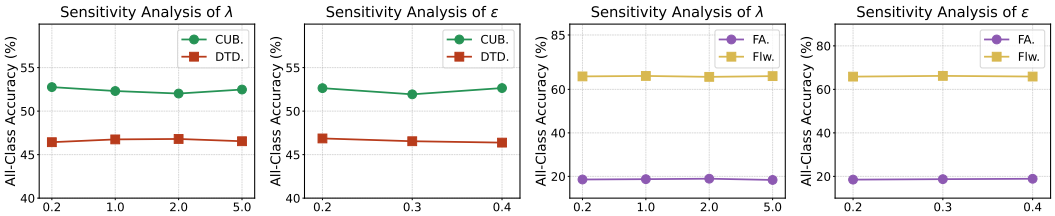


Figure 6: (a) Sensitivity analysis of λ on dataset CUB. and DTD. (b) Sensitivity analysis of ε on dataset CUB. and DTD. (c) Sensitivity analysis of λ on dataset FA. and Flw. (d) Sensitivity analysis of ε on dataset FA. and Flw.

We also perform sensitivity analysis of α (in Eq. 5) on different datasets. As shown in Fig. 7, LSR maintains relatively stable as α varies within the tested range. These results support the robustness of LSR to the choice of α. Fixing α at 0.1 is a simple and effective strategy.

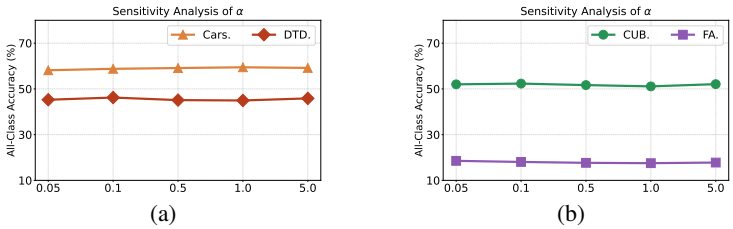


Figure 7: (a) Sensitivity analysis of α on datasets Cars. and DTD. (b) Sensitivity analysis of α on datasets CUB. and FA.

The cluster ratio denotes the ratio of the number of cluster centers (in Sec. 3.2) to the number of provided classes and is set to 2.0 by default. Table 12 suggests that low cluster ratios generally

yield poorer performance. The limited number of centers may cause omitted-class samples to share semantic anchors with provided-class samples, obscuring their semantic distinctions. Over-clustering provides additional centers, which can help separate these samples and capture semantic structures beyond the adaptation vocabulary. However, further increases in the number of centers do not consistently improve performance.

Table 13 examines the sensitivity of LSR to the anchor-filtering parameter δ (in Sec. F.2). Setting $\delta = 0.0$ disables filtering and yields slightly lower accuracy, suggesting that retaining unreliable anchors can weaken reconstruction-based supervision. Conversely, increasing δ to 0.5 also reduces performance, as overly aggressive filtering may remove useful anchors. Therefore, fixing δ to a small value of 0.05 is sufficient to achieve competitive performance.

Table 12: Sensitivity to the cluster ratios.

Dataset	0.1	0.2	1.0	2.0	3.0
DTD.	44.09	45.37	45.31	46.86	45.53
Flw.	63.21	63.66	63.35	64.53	65.19
Cars.	57.76	58.58	58.50	59.24	57.97
RES.	62.02	63.24	63.55	64.52	63.77

Table 13: Sensitivity analysis of δ.

Dataset	0.0	0.05	0.1	0.5
DTD.	45.05	45.59	45.53	45.31
CUB.	64.12	64.53	64.92	64.30
Cars.	63.45	64.52	64.25	63.51
FA.	17.88	18.55	18.39	16.23

H Compute Resources

- CPU: AMD EPYC 7642 48-Core Processor × 2
- GPU: NVIDIA GeForce RTX 4090 24G × 1
- MEM: 500G
- Maximum total computing time per dataset: training + testing $\approx$ 30min.