Cluster-Aware Neural Collapse Prompt Tuning for Long-Tailed Generalization of Vision-Language Models

Supplementary Material

6. Experiments Setting

6.1. Training Details.

We evaluate our PromptCNC under base-to-new generalization, domain generalization, and cross-dataset transfer generalization over 11 image classification benchmark datasets. We conduct the experiments based on the vision backbone with Vit-B/16. We set the number of visual and textual prompts to 4, initializing them with the template 'a photo of a []'. All our models are trained with a batch size of 4. We utilize SGD as our optimization method, starting with an initial learning rate of 0.0025 and employing a cosine annealing scheduler that includes one warm-up epoch. Depending on the specific experiment setup, the training duration varies: 1)in the base-to-new setting, models are trained for 12 epochs; 2)in the few-shot setting, the training extends to 25 epochs; 3)in domain generalization and cross-dataset evaluation, we limit training to 8 epochs. All experiments are conducted based on a single NVIDIA 3090 GPU. To simulate imbalance, we downsample each class to follow an exponential decay distribution, controlled by imbalance ratios $\tau \in \{1, 0.25, 0.6\}$. Here, τ is defined as the ratio between the smallest and largest class sizes, i.e., $\tau = \min\{n_k\}/\max\{n_k\}$, where n_k denotes the number of training samples in the k-th class. We fix $\max\{n_k\} = 16$ across all settings.

7. Ablative Analysis

7.1. Ablation on the Number of Samples per Cluster.

To investigate the sensitivity of our method to the number of samples per cluster, we conduct an ablation study by varying this value across a range of settings. This analysis explores how the granularity of semantic grouping—measured by how many class prototypes are grouped into each cluster—affects model performance and stability. In our framework, frozen textual prototypes are grouped into disjoint semantic clusters using K-means before training. The number of samples per cluster determines how coarse or fine the semantic partitioning is. Larger clusters (i.e., more samples per cluster) result in coarser partitions, enabling broader semantic sharing but potentially reducing local discriminability. In contrast, smaller clusters offer finer-grained separation, which may improve within-cluster alignment but risks overfitting due to reduced intra-cluster diversity. To better understand this trade-off, we evaluate PromptCNC under varying cluster sizes, specifically using average per-cluster sample counts of $\{4, 8, 16, 32, 64\}$ prototypes. Results in Table 5 show that performance improves as cluster size decreases, peaking at 73.92% when each cluster contains 16 prototypes.

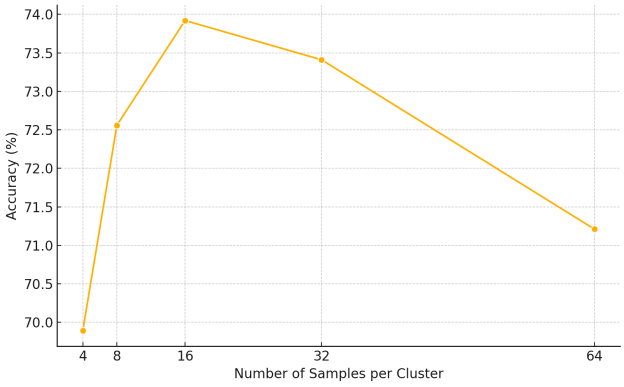

Figure 5. Effect of the number of samples per cluster on base-to-new generalization performance. Results are reported on the ImageNet dataset, where each point represents the harmonic mean (**H**) between base and novel class accuracy.

7.2. Ablation on the choice of clustering algorithm.

Our scaffold (Sec. 3.2) performs *static* clustering on frozen textual features to preserve the pre-trained semantic structure and to provide cluster-local neighborhoods for NC constraints. While the main paper uses K-means for static cluster mining, we examine whether alternative clustering algorithms affect the scaffold quality and the final performance, as shown in Table5. We evaluate four common algorithms in the ImageNet base-to-new setting: (1) Euclidean K-means, (2) Cosine K-means (spherical), (3) Spectral clustering (cosine affinity, normalized Laplacian), and (4) K-medoids (PAM, cosine distance). For fairness, we fix the cluster count M across algorithms, use the *same* random-seed grid, and tune algorithm-specific hyperparameters *only* on the frozen features via a cosine-silhouette criterion (no label or test data is used). We report two metrics: (i) base-to-new harmonic mean (H) under $\tau \in \{0.25, 0.06\}$ and (ii) seed-to-seed standard deviation (stability).

Across datasets we observe three consistent trends. First, cosine-consistent partitioners (Cosine K-means) achieve the best H on both imbalance levels, with lower run-to-run variance than Euclidean K-means. Second, Spectral clustering is competitive but more sensitive to initialization and the affinity scale, yielding slightly lower H and higher variance. Third, K-medoids tends to underperform in both H and sta-

bility, indicating that medoid-only representatives provide a weaker scaffold for our cluster-local NC losses. These results support our choice of a cosine-aligned partitioner for the static scaffold while confirming that the gains of PromptCNC do not rely on a specific clustering heuristic.