# SAAS: SIZE-AWARE ANCHOR SELECTION FOR GENERALIZED CATEGORY DISCOVERY

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Generalized category discovery (GCD) assigns unlabeled images to known and novel categories using supervision from known classes. Reliable groups of similar images can help correct category predictions, but assessing their reliability through graph density can penalize larger groups. This bias arises because sparse nearest-neighbor graphs limit the connections available to each image, whereas complete-graph normalization counts all possible connections within a group. We propose Size-Aware Anchor Selection (SAAS), a general reliability assessment strategy for selecting reliable local groups under sparse graph constraints, instantiated as a post-hoc prediction correction framework without additional training. SAAS normalizes the observed connections by the maximum number permitted by the group size and neighborhood size. It selects groups whose images are similar and whose membership remains consistent across two augmented views, using these groups as anchors for correction. Similarity to the selected groups guides corrections to ambiguous predictions. We evaluate SAAS on six generic and fine-grained GCD benchmarks using DINOv2 pretraining. On CUB, the resulting pipeline achieves 88.67% overall accuracy, with 85.80% on known classes and 90.10% on novel classes. Ablation studies show that removing the group-size cap alone leaves predictions unchanged, whereas the proposed normalization improves accuracy and enables further gains when the cap is removed. These results highlight the importance of accounting for sparse connectivity when selecting reliable groups for GCD prediction correction.

## 1 INTRODUCTION

Generalized category discovery (GCD) aims to recognize known categories and discover novel ones in partially labeled datasets, where unlabeled images may belong to either group (Vaze et al., 2022a). Existing methods learn visual representations and assign categories through clustering or a parametric classifier (Wen et al., 2023; Zhao et al., 2023; Choi et al., 2024; Liu et al., 2025b; An et al., 2023; Cao et al., 2024). In clustering-based pipelines, samples lying at similar distances from competing cluster centers can have ambiguous assignments. Local relations among samples provide complementary evidence for refining these predictions (Zhou et al., 2003; Iscen et al., 2019), motivating the use of local groups of similar images as reference anchors for correcting predictions.

The reliability of these groups should be assessed according to the neighborhood graph from which they are constructed. A mutual-$q$ nearest-neighbor (mutual-$q$NN) graph retains an edge only when two samples select each other, limiting each sample to at most $q$ connections (Maier et al., 2009). Conventional complete-graph density normalizes internal connectivity by the number of all possible sample pairs. For sufficiently large groups, this denominator grows quadratically with group size, whereas the maximum internal connectivity permitted by the graph grows only linearly. This mismatch can give larger groups lower density scores even when they use a similar fraction of their available connections. A fixed density threshold can consequently exclude coherent groups because of their size.

We propose **Size-Aware Anchor Selection (SAAS)**, a method that selects reliable local groups to correct GCD predictions after clustering, without additional training (Figure 1). Building on the non-parametric Hyp-SelEx variant of HypCD (Liu et al., 2025b), we construct candidate groups within clusters defined by the initial clustering assignments, termed uncorrected predictions. Our



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.

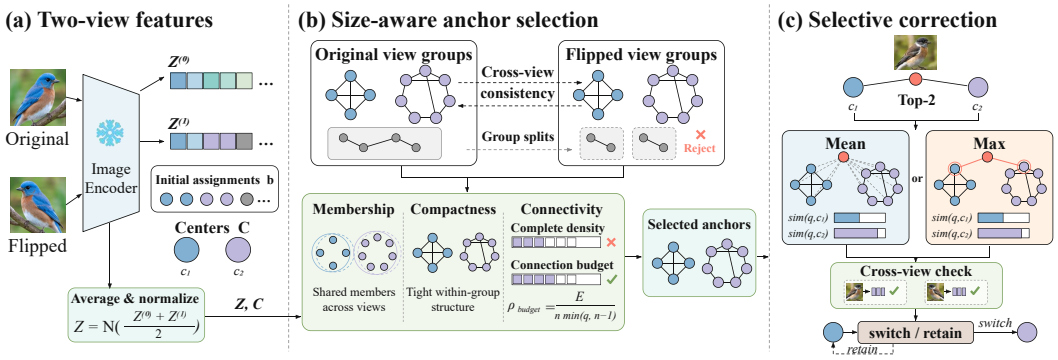



Figure 1: Overview of SAAS and its anchor-based prediction correction pipeline. HypCD features from original and flipped images yield cross-view-stable anchor groups selected by connection-budget normalization. Mean or maximum member similarity then guides selective correction between the two nearest cluster centers.

reliability criterion normalizes internal connectivity by the connection budget: the upper bound on internal connectivity imposed by group size and neighborhood size. We also require high within-group feature similarity (compactness) and consistent membership between original and horizontally flipped views (cross-view stability). Selected groups have no fixed group-size cap (upper size limit).

For each ambiguous sample, termed a query, anchor support measures its similarity to selected groups in each of the two nearest-center categories. We compare these categories when both have eligible anchors (two-sided support). Support uses mean or maximum query–member similarity; stronger support for the second category proposes a switch. A query-level cross-view check requires the query's neighbors in both views to support that category. Corrections are applied once using the original features and predictions.

Experiments on six GCD benchmarks with DINOv2 pretraining evaluate correction gains; CUB ablations isolate normalization and group-size effects, followed by analyses of anchor support, correction quality, aggregation, and neighborhood size.

Our contributions are summarized as follows: (1) We identify a size-dependent mismatch between complete-graph density normalization and the bounded connectivity induced by sparse neighborhood graphs in local-group reliability assessment. (2) We develop a connection-budget-aware reliability assessment framework for local groups, which enables anchor-based selective prediction correction. (3) Ablations and paired comparisons show that the proposed normalization recovers larger reliable groups, expands two-sided anchor support, and improves correction outcomes; experiments across six GCD benchmarks examine its transfer beyond CUB.

## 2 Related Work

**Generalized category discovery.** Building on novel class discovery (Han et al., 2019; 2020; Fini et al., 2021; Zhong et al., 2021), the GCD baseline (Vaze et al., 2022a) combines DINO features (Caron et al., 2021), contrastive learning (Chen et al., 2020; Khosla et al., 2020), and semi-supervised $k$-means. Later methods explore concept learning (Pu et al., 2023), parametric classification (Wen et al., 2023; Liu et al., 2025a), self-coding (Rastegar et al., 2023), and debiasing (Liu & Han, 2025; Cao et al., 2022; Guo et al., 2025). SPTNet uses spatial prompts (Wang et al., 2024), SelEx learns hierarchical self-expertise (Rastegar et al., 2024), HypCD adopts hyperbolic geometry (Liu et al., 2025b), and HiLo disentangles semantic and domain features (Wang et al., 2025).

**Neighborhood graphs and group reliability.** FINCH and CiPR form neighbor-based hierarchies (Sarfraz et al., 2019; Hao et al., 2024); NNCLR and PromptCAL exploit neighbors and affinities for contrastive learning (Dwibedi et al., 2021; Zhang et al., 2023). Graph construction (Maier et al., 2009; von Luxburg, 2007; Zelnik-manor & Perona, 2004) and resampling stability (Hennig, 2007) inform group reliability. We account for sparse connectivity constraints when selecting anchors for post-hoc GCD correction.

# 3 METHOD

**Problem statement.** Given labeled data $\mathcal{D}_L$ from known categories and unlabeled data $\mathcal{D}_U$ from known and novel categories, GCD uses $\mathcal{D}_L$ to recognize and discover categories in $\mathcal{D}_U$. We assume the total number of categories $K$ is known.SAAS assesses local-group reliability after clustering and uses selected reliable groups to refine ambiguous predictions.

## 3.1 BACKGROUND: HYPCD

HypCD (Liu et al., 2025b) introduces hyperbolic representation learning into both parametric and non-parametric generalized category discovery pipelines. We adopt Hyp-SelEx, its non-parametric variant that extends hierarchical self-expertise to hyperbolic space. Let $f$ denote the backbone network and $g$ the projection head. For an augmented training image $x_i$, the projected feature is clipped before being mapped to the Poincaré ball:

$$e_i = g(f(x_i)), \qquad h_i = \exp_o^c(\mathrm{clip}_r(e_i)), \tag{1}$$

where $\mathrm{clip}_r$ leaves features with norm at most $r$ unchanged and rescales larger features to norm $r$. The exponential map $\exp_o^c$ maps the clipped feature to the Poincaré ball at base point $o$, with curvature parameter $c$. Hyp-SelEx combines unsupervised self-expertise for hard-negative discrimination with supervised self-expertise based on hierarchical pseudo-labels. Both use distance- and angle-based objectives in hyperbolic space; we retain the Hyp-SelEx training objective.

After training, Hyp-SelEx retains the label assignment strategy of SelEx, applying semi-supervised $k$-means directly to feature representations extracted by the backbone in Euclidean space (Liu et al., 2025b). Our anchor construction and prediction correction also operate on backbone features. For $N$ samples and $K$ categories, we denote the two-view features by $Z^{(0)}, Z^{(1)} \in \mathbb{R}^{N\times d}$, the scoring features by $Z \in \mathbb{R}^{N\times d}$, and the cluster centers by $C \in \mathbb{R}^{K\times d}$, where $d$ is the backbone feature dimension. We refer to the clustering output before correction as the *uncorrected predictions*, denoted by $b \in \{1,\dots,K\}^N$. The *corrected predictions* $\hat{b} \in \{1,\dots,K\}^N$ denote the complete output after correction, including entries that remain unchanged. The two-view features are used for graph construction and group reliability assessment, whereas the scoring features are used to compute cluster-center distances and anchor support. Feature extraction and view aggregation are specified in Section 4.1.

When a sample lies at similar distances from competing cluster centers, center-based evidence may not clearly distinguish the candidate categories. We therefore construct reliable local anchor groups from the representations and uncorrected predictions, and use their support as complementary evidence for selectively correcting ambiguous predictions. Throughout correction, the encoder, extracted features, and cluster centers remain fixed; only predictions are updated.

## 3.2 CONNECTION-BUDGET-AWARE GROUP RELIABILITY

In each view $v \in \{0,1\}$, we retain an edge when two samples select each other among their $q$ nearest neighbors (Maier et al., 2009). Let $A^{(v)} \in \{0,1\}^{N\times N}$ be the symmetric adjacency matrix, with $A_{ii}^{(v)} = 0$. For a group $G \subseteq \{1,\dots,N\}$ of size $n = |G| \geq 2$, the internal connectivity $E_v(G) = \sum_{i,j\in G} A_{ij}^{(v)}$ counts each undirected edge twice, and the complete-graph density is $\rho_v^{\mathrm{full}}(G) = E_v(G)/[n(n-1)]$.

**Proposition 1 (Implicit size ceiling).** In a mutual-$q$NN graph, a complete-density threshold $\tau_\rho \in (0,1]$ cannot be satisfied by any group with $n > 1 + q/\tau_\rho$, even if every vertex exhausts its internal degree capacity. The proof is provided in Appendix D.

To account for this size dependence, we normalize internal connectivity by the degree-capacity bound, since each member has at most $\min(q, n-1)$ neighbors within the group:

$$\rho_v^{\mathrm{budget}}(G) = \frac{E_v(G)}{n\min(q, n-1)}. \tag{2}$$

This score lies in $[0,1]$ and measures utilization of the group's connection budget. It equals $\rho_v^{\mathrm{full}}(G)$ for $n \leq q+1$; otherwise, its denominator is $nq$, growing linearly with group size. For $q = 10$ and

$\tau_\rho = 0.5$, groups with $n > 21$ cannot pass the complete-density threshold; Figure 2 illustrates this limit and compares the two normalizations on a 26-sample CUB group.

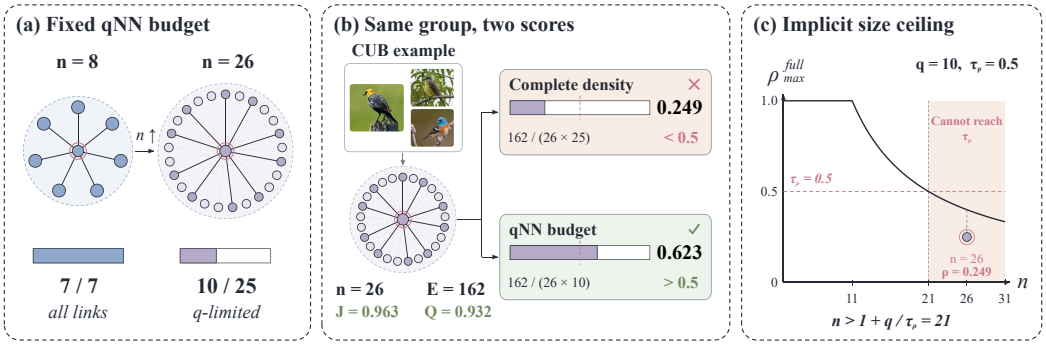



Figure 2: Connection-budget normalization with $q = 10$; links are schematic. (a) Per-vertex connection budgets. (b) The same 26-sample CUB group has complete-graph and budget densities of 0.249 and 0.623, with $E = \min_v E_v(G) = 162$ counting each undirected edge twice; $J$ measures cross-view membership overlap and $Q$ measures within-group feature similarity. (c) Complete-graph density cannot reach 0.5 for $n > 21$.

### 3.3 Anchor Group Construction

We construct candidate anchors from connected components within clusters defined by the uncorrected predictions. For each view $v$, we build the mutual-$q$NN graph using $\ell_2$-normalized feature similarities. For cluster $k$, with $I_k = \{i : b_i = k\}$, we collect the connected components of the induced subgraph as $\mathcal{G}_k^{(v)}$ and define $\mathcal{G}^{(v)} = \bigcup_{k=1}^{K} \mathcal{G}_k^{(v)}$. Components from view 0 serve as candidates, while those from view 1 provide the reference partition. Each candidate $G \in \mathcal{G}_k^{(0)}$ inherits label $c(G) = k$.

Cluster-wise stability can be assessed by Jaccard matching between a cluster and its most similar counterpart under data perturbations (Hennig, 2007). We apply this membership-based comparison across the two image views and complement it with a feature-compactness measure. For a candidate $G$ of size $n = |G| \geq 2$, we define

$$J(G) = \max_{H \in \mathcal{G}^{(1)}} \frac{|G \cap H|}{|G \cup H|},$$
$$Q(G) = \min_{v \in \{0,1\}} \frac{1}{n(n-1)} \sum_{\substack{i,j \in G \\ i \neq j}} \left\langle z_i^{(v)}, z_j^{(v)} \right\rangle. \tag{3}$$

Here, $H$ is a reference-view component, and $J(G)$ measures the strongest membership overlap with that partition. The compactness score $Q(G)$ is the smaller of the two view-wise mean pairwise similarities, so a high score requires compact features in both views. We evaluate connectivity for the same candidate members in both view graphs, using $\rho^{\text{budget}}(G) = \min_{v \in \{0,1\}} \rho_v^{\text{budget}}(G)$ from Eq. (2). Let $n_{\min} \geq 2$ be the minimum group size, and let $\tau_J$, $\tau_Q$, and $\tau_\rho$ be the thresholds for cross-view stability, feature compactness, and budget-normalized connectivity, respectively. The anchor set is

$$\mathcal{A} = \left\{ G \in \mathcal{G}^{(0)} \,\middle|\, \begin{array}{ll} |G| \geq n_{\min}, & J(G) \geq \tau_J, \\ Q(G) \geq \tau_Q, & \rho^{\text{budget}}(G) \geq \tau_\rho \end{array} \right\}. \tag{4}$$

Group size is constrained only from below, allowing large components that meet the reliability thresholds to contribute anchor evidence.

### 3.4 Anchor-Based Prediction Correction

We identify ambiguous predictions from the relative distance gap between the two nearest cluster centers and use anchor support to compare the corresponding candidate classes. Let $\mathbf{c}_k \in \mathbb{R}^d$ denote

the $k$-th center in the cluster-center matrix $C$, and let $z_i \in \mathbb{R}^d$ be the scoring feature of sample $i$. The indices of its nearest and second-nearest centers are $k_{i,1}$ and $k_{i,2}$, with squared distances $d_{i,r} = \|z_i - \mathbf{c}_{k_{i,r}}\|_2^2$ for $r \in \{1, 2\}$ and $d_{i,1} \leq d_{i,2}$. We retain the original clustering assignment $b_i$ and measure ambiguity using the relative distance margin

$$m_i = \frac{d_{i,2} - d_{i,1}}{\max(d_{i,2}, \epsilon)}, \tag{5}$$

where $\epsilon = 10^{-12}$ stabilizes the denominator. A small $m_i$ indicates similar distances to the two competing centers. With ambiguity threshold $\tau_m$, a query is eligible for correction when $m_i \leq \tau_m$ and $b_i = k_{i,1}$, so that its original prediction is the first candidate. All other queries retain $b_i$.

For each selected query $i$, we exclude all anchor groups containing $i$ (self-group exclusion) and compute support using normalized scoring features $\bar{z}_i = z_i / \|z_i\|_2$. The support of an eligible group $G$ is summarized by either its mean member similarity or its strongest member similarity:

$$\begin{aligned} a_{\text{mean}}(i, G) &= \frac{1}{|G|} \sum_{j \in G} \bar{z}_i^\top \bar{z}_j, \\ a_{\max}(i, G) &= \max_{j \in G} \bar{z}_i^\top \bar{z}_j. \end{aligned} \tag{6}$$

Let $a(i, G)$ denote the chosen within-group aggregation. Let $\mathcal{A}_k(i) = \{G \in \mathcal{A} : c(G) = k,\ i \notin G\}$ denote the eligible anchor set for candidate class $k$. We compare the two candidate classes only when both $\mathcal{A}_{k_{i,1}}(i)$ and $\mathcal{A}_{k_{i,2}}(i)$ are nonempty after self-group exclusion; this is two-sided anchor support. Otherwise, we retain the uncorrected prediction. For a nonempty eligible anchor set $\mathcal{A}_k(i)$, the class support is defined as

$$s(i, k) = \max_{\substack{G \in \mathcal{A} \\ c(G) = k,\ i \notin G}} a(i, G). \tag{7}$$

This outer maximum selects the strongest eligible anchor group associated with category $k$ under either aggregation rule. For an eligible query with two-sided anchor support, a switch to $k_{i,2}$ is proposed when $s(i, k_{i,2}) > s(i, k_{i,1})$. Before applying the switch, we additionally require fixed query-level neighborhood evidence from the two views to support the target category. If this confirmation fails, the uncorrected prediction $b_i$ is retained. All correction decisions are computed from the fixed uncorrected predictions, anchor groups, and neighborhoods and are applied in a single pass; corrected predictions are not reused for subsequent decisions.

# 4 Experiments

## 4.1 Experimental Setup

**Datasets.** Our experiments include three generic classification datasets, CIFAR-10 and CIFAR-100 (Krizhevsky & Hinton, 2009), and ImageNet-100 (Russakovsky et al., 2015), and three fine-grained datasets from the Semantic Shift Benchmark (SSB) (Vaze et al., 2022b): CUB (Wah et al., 2011), Stanford Cars (Krause et al., 2013), and FGVC-Aircraft (Maji et al., 2013). We follow the official known/novel class splits used in prior GCD studies (Vaze et al., 2022a). For CUB, the main comparison and correction-quality analysis use the 4,496 unlabeled images from the official training split, which are also used as unlabeled data during encoder training. Using the same encoder, we conduct the normalization and group-size-cap ablations and neighborhood-size sensitivity analysis on the disjoint official test split of 5,794 images for development analyses, including the selection of $q$ described below.

**Evaluation metrics.** Following previous GCD works (Vaze et al., 2022a; Wen et al., 2023), we report All, Old, and New accuracy. Given predicted assignments $\hat{y}$ and ground-truth labels $y$, we first obtain the optimal Hungarian matching $\pi^*$:

$$\pi^* = \arg\max_{\pi} \sum_i \mathbb{I}(y_i = \pi(\hat{y}_i)). \tag{8}$$

The matched predictions are then used to compute All accuracy over all samples, and Old/New accuracy over known/novel categories, respectively.

For the standard GCD comparison in Table 1, Hungarian matching is computed independently for each prediction output, following the conventional evaluation protocol. For correction statistics and paired ablations, we instead obtain the mapping from the uncorrected predictions and keep it fixed across correction variants. Evaluation labels are used for matching and outcome measurement, not for anchor construction or individual correction decisions. We report paired accuracy changes relative to each run's uncorrected predictions, with $\Delta$All denoting the All change in percentage points. We count wrong-to-correct fixes $F$, correct-to-wrong harms $H$, and wrong-to-wrong changes $W$. The total number of modifications is $M = F + H + W$, correction precision is $F/M$, and net correction is $F - H$. Modification coverage is $M/N$, where each seed–image event is one image evaluated in one clustering run. Thus, the 4,496-image evaluation over three seeds has $N = 13{,}488$. Original candidate scope lets each method use its own supported queries; common support restricts updates to queries supported by every compared method. Both retain denominator $N$.

**Compared methods.** The main comparison includes GCD (Vaze et al., 2022a), SimGCD (Wen et al., 2023), and the Hyp-GCD and Hyp-SimGCD variants of HypCD (Liu et al., 2025b). Their published DINOv2 results are taken from Liu et al. (2025b).Alongside these published results, we report our Hyp-SelEx results and those obtained with prediction correction using SAAS. The analysis uses the uncorrected predictions as a paired baseline and compares complete-density and size-aware anchor selection, mean and maximum aggregation (the size-aware mean/max variants of SAAS), KNN correction, and nearest-member correction. KNN requires at least six of ten neighbors to support the second candidate; nearest-member correction compares the strongest member similarity in the two candidate clusters.

**Implementation details.** We adopt Hyp-SelEx (Liu et al., 2025b) with a DINOv2-pretrained ViT-B/14 backbone (Dosovitskiy et al., 2021; Oquab et al., 2024), assuming a known number of categories. We extract 768-dimensional CLS-token features after the final LayerNorm, before projection and hyperbolic mapping. Original and horizontally flipped views are encoded and $\ell_2$-normalized separately to obtain $Z^{(0)}$ and $Z^{(1)}$. Their row-wise average is $\ell_2$-normalized to form $Z$. Individual-view features construct the graphs, while $Z$ provides uncorrected clustering predictions, center distances, and anchor-support scores (Section 3.4). Paired comparisons share the extracted features and uncorrected predictions.

All six datasets share the anchor-selection thresholds $n_{\min} = 3$, $\tau_J = 0.8$, $\tau_Q = 0.8$, and $\tau_\rho = 0.5$, with reliability criteria satisfied in both views and no upper group-size bound. We use $q = 10$ and $\tau_m = 0.2$ throughout; $q$ is selected based on the neighborhood-size analysis on the disjoint 5,794-image CUB split, while $\tau_m$ follows the existing correction setup. The same query-level cross-view confirmation rule is used for all datasets. Correction is applied once, with mean and maximum within-group aggregation compared under otherwise identical settings.

### 4.2 Main Comparison

**CUB results.** Table 1 reports the Hyp-SelEx baseline (86.6500/84.3200/87.8200 on All/Old/New). The mean and maximum variants both achieve 88.6655/85.8039/90.0968. Under the standard independently matched GCD evaluation, their All accuracies exceed the Hyp-SelEx baseline by 2.0155 percentage points. Additional transfer experiments applying SAAS to predictions from an alternative GCD pipeline are provided in Appendix H.

**Additional completed datasets.** Each correction is paired with the corresponding uncorrected predictions. All six datasets use $q = 10$ and $\tau_m = 0.2$. These configurations yield five-seed average All accuracy changes of $+0.2896$, $-0.0130$, $+0.0800$, $-0.0240$, and $+0.3872$ percentage points on CIFAR-10, Cars, CIFAR-100, Aircraft, and ImageNet-100, respectively.

Table 1: Comparison on six datasets with DINOv2 pretraining. ACC (%) is reported for All, Old, and New classes. Light-blue rows report SAAS results; mean/max denote mean or maximum within-group anchor aggregation. The five additional datasets report five-seed averages.

| | CUB | | | Stanford Cars | | | FGVC-Aircraft | | | CIFAR-10 | | | CIFAR-100 | | | ImageNet-100 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Method | All | Old | New | All | Old | New | All | Old | New | All | Old | New | All | Old | New | All | Old | New |
| GCD | 71.9 | 71.2 | 72.3 | 65.7 | 67.8 | 64.7 | 55.4 | 47.9 | 59.2 | 97.8 | 99.0 | 97.1 | 79.6 | 84.5 | 69.9 | 78.8 | 89.5 | 73.0 |
| SimGCD | 71.5 | 78.1 | 68.3 | 71.5 | 81.9 | 66.6 | 63.9 | 69.9 | 60.9 | 98.7 | 96.7 | 99.7 | 88.5 | 89.2 | 87.2 | 89.9 | 95.5 | 87.1 |
| Hyp-GCD | 75.6 | 75.1 | 75.9 | 72.8 | 80.4 | 69.1 | 62.7 | 70.0 | 59.0 | 97.5 | 98.9 | 96.8 | 84.5 | 87.5 | 78.5 | 82.9 | 92.4 | 78.2 |
| Hyp-SimGCD | 77.6 | 77.9 | 77.4 | 82.5 | 85.8 | 81.0 | 76.4 | 70.3 | 79.4 | 98.9 | 97.7 | 99.5 | 91.5 | 90.0 | 94.6 | 91.9 | 96.2 | 89.8 |
| Hyp-SelEx* | 86.7 | 84.3 | 87.8 | 88.3 | 89.9 | 87.5 | 82.1 | 80.8 | 82.8 | 98.0 | 97.7 | 98.1 | 88.8 | 90.2 | 85.9 | 82.6 | 96.1 | 75.8 |
| **SAAS* (mean)** | **88.7** | **85.8** | **90.1** | **88.3** | **89.9** | **87.5** | **82.1** | **80.8** | **82.8** | **98.3** | **97.7** | **98.5** | **88.9** | **90.2** | **86.2** | **83.0** | **96.2** | **76.4** |
| **SAAS* (max)** | **88.7** | **85.8** | **90.1** | **88.3** | **89.9** | **87.5** | **82.1** | **80.8** | **82.8** | **98.3** | **97.7** | **98.5** | **88.9** | **90.2** | **86.2** | **83.0** | **96.2** | **76.4** |

*results from our implementation.

### 4.3 Analysis

Table 2: Factorized ablation on 5,794 development images. All/Old/New report mean accuracy (%). Complete/Budget denote complete-graph/connection-budget normalization. Cap limits anchor-group size (None: unlimited); Score specifies mean/max query–member similarity within each group. M/F/H are modifications/fixes/harms summed over three clustering seeds. Appendix C.1 reports run-to-run variability.

| Normalization | Cap | Score | All | Old | New | M/F/H |
|---|---|---|---|---|---|---|
| Uncorrected predictions | – | – | 84.4264 | 85.4716 | 83.3906 | 0/0/0 |
| Complete | 20 | mean | 84.4207 | 85.4716 | 83.3792 | 1/0/1 |
| Complete | 20 | max | 84.4207 | 85.4716 | 83.3792 | 1/0/1 |
| Complete | None | mean | 84.4207 | 85.4716 | 83.3792 | 1/0/1 |
| Complete | None | max | 84.4207 | 85.4716 | 83.3792 | 1/0/1 |
| Budget | 20 | mean | 84.4552 | 85.5525 | 83.3677 | 20/9/4 |
| Budget | 20 | max | 84.4955 | 85.5987 | 83.4021 | 21/13/1 |
| Budget | None | mean | 84.5357 | 85.8414 | 83.2417 | 87/39/20 |
| Budget | None | max | 84.5818 | 85.7605 | 83.4135 | 85/43/16 |

**Normalization and group-size cap.** Table 2 ablates normalization and the group-size cap on the separate 5,794-image CUB development evaluation at $q = 10$. The paired variants share the same features, uncorrected predictions, and fixed Hungarian mapping.

With a cap of 20, replacing complete-graph normalization with connection-budget normalization raises All accuracy to 84.4552% (mean) and 84.4955% (max). Removing the cap further raises these values to 84.5357% and 84.5818%, respectively, whereas removing it under complete-graph normalization leaves predictions unchanged. This indicates that, in this setting, the limiting factor is the normalization-induced size penalty of complete-graph density.

**Effect on anchor support.** Figure 3 shows how normalization and the size cap affect retained groups. At cap 20, budget normalization increases the mean number of groups of size 12–20 from 8 to 42. Removing the cap additionally retains 103 groups of size 21–30 and 6.67 larger groups per run on average. With budget normalization and maximum aggregation, removing the group-size cap expands two-sided support from 39 to 200 query events. Both configurations yield identical final predictions on queries supported by both configurations, while most newly supported queries retain their uncorrected predictions. The additional net improvement therefore comes entirely from newly supported queries, linking the larger groups retained in Fig. 3 to the paired improvement in

Table 2. Figure 4 illustrates the resulting switch and retain decisions. Correction outcomes and per-run results appear in Appendices A.3 and C.2.

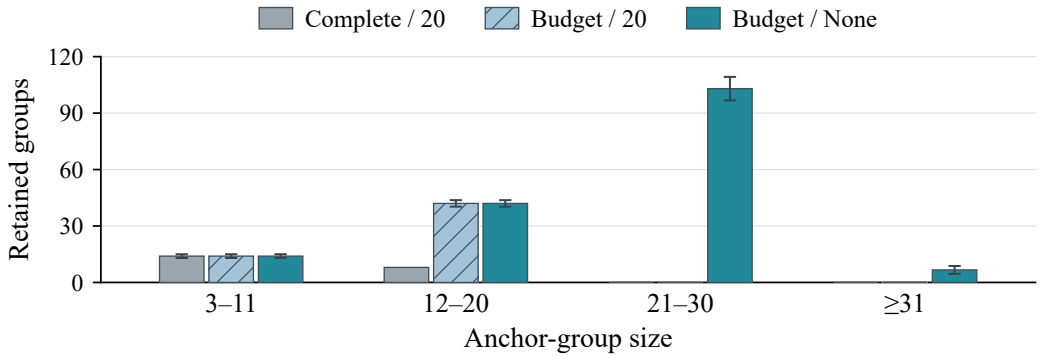



Figure 3: Anchor-group size distribution on the 5,794-image CUB development evaluation at $q = 10$. Bars and error bars show the mean and sample standard deviation of retained-group counts over three seeds. The legend specifies normalization and the group-size cap (None: no upper limit). Configurations share candidate groups and all remaining selection criteria. Complete / None selects the same groups as Complete / 20 in every run.

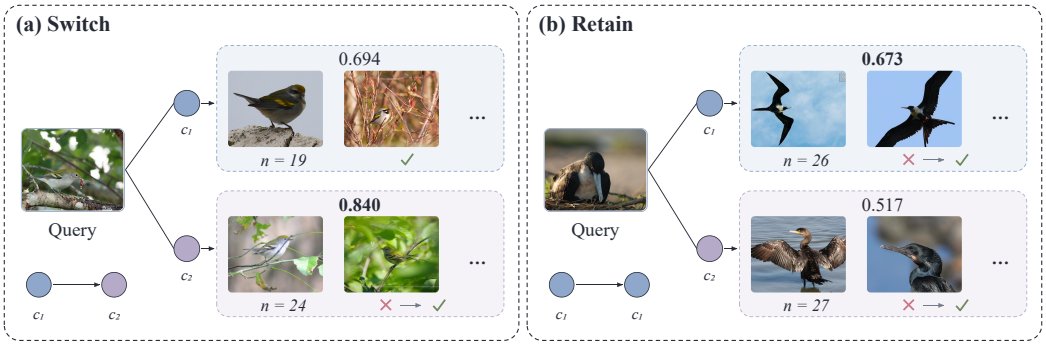



Figure 4: Removing the size cap enables two-sided anchor support: (a) switch; (b) retain. Photographs show example members of the selected groups, with $n$ denoting group size.

**Coverage and correction quality.** To isolate prediction-level correction effects from changes in the optimal Hungarian permutation, Table 3 keeps the mapping obtained from the uncorrected predictions fixed. Table 3 reports outcomes on 4,496 CUB images. In the original candidate scope, mean and maximum aggregation achieve correction precisions of **68.42%** and **60.00%**, respectively, compared with 35.71% for KNN and 36.11% for nearest-member correction.

On the 49 common-support events, KNN and nearest-member correction yield net corrections of +7 and +12, respectively, while the anchor results remain unchanged. SAAS yields positive net correction with fewer modifications and harms in the original scope; improved KNN and nearest-member results under common support highlight the influence of candidate scope.

**Parameter sensitivity.** On the 5,794-image CUB development split, Figure 5(a) shows positive average All gains for $q \in \{5, 10, 15, 20\}$, peaking at $q = 10$, with maximum aggregation and $\tau_m = 0.2$. Complete-density selection remains near the uncorrected baseline. One run at $q = 5$ has negative net correction; average New gains are negative at $q = 5, 15, 20$ (Appendices A.1 and C.3).

At $q = 10$, both aggregation rules yield positive average All gains across $\tau_m \in \{0.05, 0.1, 0.2, 0.3\}$ (Figure 5(b)). Eligible queries increase from 0.90% to 11.30%, and eligible queries with two-

Table 3: Correction outcomes on the same 4,496 CUB images ($N = 13{,}488$ seed–image events). $M/F/H/W$ denote modified, wrong-to-correct, correct-to-wrong, and wrong-to-wrong events. Common support contains 49 shared events; coverage retains denominator $N$, and Net is $F - H$.

| Method | $M$ | M/N (%) | $F$ | $H$ | $W$ | F/M (%) | Net |
|---|---|---|---|---|---|---|---|
| *Original candidate scope* | | | | | | | |
| Uncorrected predictions | 0 | 0.000 | 0 | 0 | 0 | – | +0 |
| KNN | 70 | 0.519 | 25 | 29 | 16 | 35.71 | -4 |
| Nearest member | 108 | 0.801 | 39 | 42 | 27 | 36.11 | -3 |
| Size-aware mean | 19 | 0.141 | 13 | 3 | 3 | 68.42 | +10 |
| Size-aware max | 15 | 0.111 | 9 | 3 | 3 | 60.00 | +6 |
| *Common support* | | | | | | | |
| Uncorrected predictions | 0 | 0.000 | 0 | 0 | 0 | – | +0 |
| KNN | 13 | 0.096 | 10 | 3 | 0 | 76.92 | +7 |
| Nearest member | 20 | 0.148 | 12 | 0 | 8 | 60.00 | +12 |
| Size-aware mean | 19 | 0.141 | 13 | 3 | 3 | 68.42 | +10 |
| Size-aware max | 15 | 0.111 | 9 | 3 | 3 | 60.00 | +6 |

sided anchor support from $0.17\%$ to $2.06\%$. Mean and maximum aggregation peak at $0.3$ and $0.2$, respectively; the existing $\tau_m = 0.2$ remains unchanged.

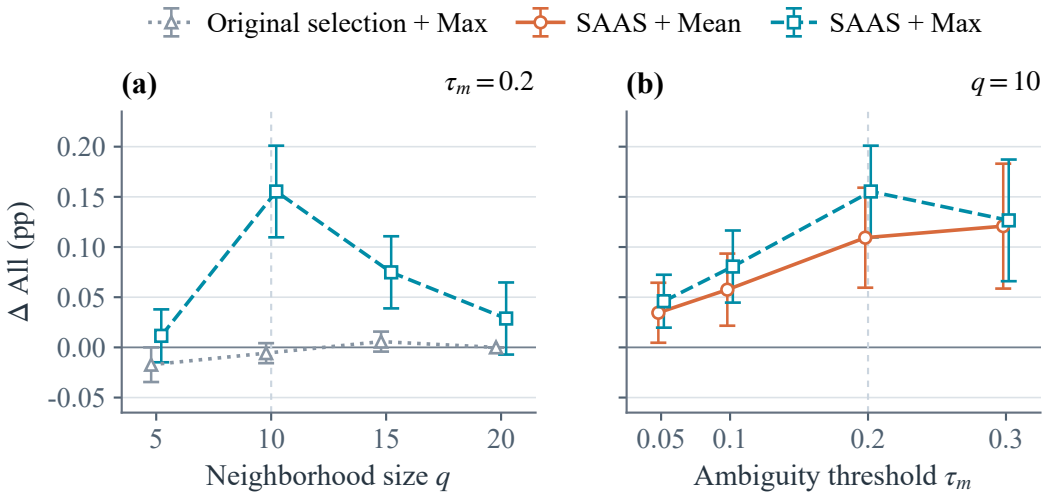



Figure 5: CUB development sensitivity to (a) neighborhood size $q$ and (b) ambiguity threshold $\tau_m$. Points show mean paired ΔAll.

Correction requires the correct category among the two nearest centers and discriminative anchor support. High connectivity and cross-view consistency do not guarantee semantic correctness, so structurally coherent groups may still support an incorrect category.

## 5 Conclusion

SAAS aligns local-group reliability assessment with the connectivity capacity of sparse neighborhood graphs. This removes the implicit size penalty of complete-graph normalization and enables larger coherent groups to provide anchor evidence. Experiments and paired analyses show that the resulting anchor support improves post-hoc GCD prediction correction without updating the encoder or cluster centers.

### AI USE STATEMENT

We used AI to aid or polish writing, for retrieval and discovery (e.g., finding related work), and for research ideation or execution. Details are described in the paper.

### REPRODUCIBILITY STATEMENT

The method section specifies graph construction, group selection, score aggregation, self-group exclusion, and the correction rule. The experimental setup records the evaluation and selection history, fixed mapping, clustering seeds, and bootstrap procedure. Appendix C reports complete results, including unfavorable outcomes and neighborhood-size analyses.

## REFERENCES

Wenbin An, Feng Tian, Qinghua Zheng, Wei Ding, QianYing Wang, and Ping Chen. Generalized category discovery with decoupled prototypical network, 2023. URL https://arxiv.org/abs/2211.15115.

Kaidi Cao, Maria Brbic, and Jure Leskovec. Open-world semi-supervised learning, 2022. URL https://arxiv.org/abs/2102.03526.

Xinzi Cao, Xiawu Zheng, Guanhong Wang, Weijiang Yu, Yunhang Shen, Ke Li, Yutong Lu, and Yonghong Tian. Solving the catastrophic forgetting problem in generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16880–16889, June 2024.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9650–9660, October 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1597–1607. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/chen20j.html.

Sua Choi, Dahyun Kang, and Minsu Cho. Contrastive mean-shift learning for generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23094–23104, June 2024.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021. URL https://arxiv.org/abs/2010.11929.

Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9588–9597, October 2021.

Enrico Fini, Enver Sangineto, Stéphane Lathuilière, Zhun Zhong, Moin Nabi, and Elisa Ricci. A unified objective for novel class discovery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9284–9292, October 2021.

Pengcheng Guo, Yonghong Song, and Boyu Wang. Towards debiased generalized category discovery. In James Kwok (ed.), *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pp. 5271–5279. International Joint Conferences on Artificial Intelligence Organization, 8 2025. doi: 10.24963/ijcai.2025/587. URL https://doi.org/10.24963/ijcai.2025/587. Main Track.

Kai Han, Andrea Vedaldi, and Andrew Zisserman. Learning to discover novel visual categories via deep transfer clustering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Kai Han, Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Andrea Vedaldi, and Andrew Zisserman. Automatically discovering and learning new visual categories with ranking statistics. In *International Conference on Learning Representations*, 2020.

Shaozhe Hao, Kai Han, and Kwan-Yee K. Wong. Cipr: An efficient framework with cross-instance positive relations for generalized category discovery, 2024. URL https://arxiv.org/abs/2304.06928.

Christian Hennig. Cluster-wise assessment of cluster stability. *Computational Statistics & Data Analysis*, 52(1):258–271, 2007. ISSN 0167-9473. doi: https://doi.org/10.1016/j.csda.2006.11.025. URL https://www.sciencedirect.com/science/article/pii/S0167947306004622.

Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Label propagation for deep semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 18661–18673. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/d89a66c7c80a29b1bdbab0f2a1a94af8-Paper.pdf.

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 (ICCV) Workshops*, pp. 554–561, June 2013.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, Toronto, Ontario, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Duo Liu, Zhiquan Tan, Linglan Zhao, Zhongqiang Zhang, Xiangzhong Fang, and Weiran Huang. Generalized category discovery via reciprocal learning and class-wise distribution regularization, 2025a. URL https://arxiv.org/abs/2506.02334.

Yuanpei Liu and Kai Han. DebGCD: Debiased learning with distribution guidance for generalized category discovery. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 48323–48336, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/789fc893d4b6e5b384f4eb058a35e31d-Paper-Conference.pdf.

Yuanpei Liu, Zhenqi He, and Kai Han. Hyperbolic category discovery, 2025b. URL https://arxiv.org/abs/2504.06120.

Markus Maier, Matthias Hein, and Ulrike von Luxburg. Optimal construction of k-nearest-neighbor graphs for identifying noisy clusters. *Theoretical Computer Science*, 410(19):1749–1764, April 2009. ISSN 0304-3975. doi: 10.1016/j.tcs.2009.01.009. URL https://www.sciencedirect.com/science/article/pii/S0304397509000590.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft, 2013. URL https://arxiv.org/abs/1306.5151.

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, 2024. URL https://arxiv.org/abs/2304.07193.

Nan Pu, Zhun Zhong, and Nicu Sebe. Dynamic conceptional contrastive learning for generalized category discovery, 2023. URL https://arxiv.org/abs/2303.17393.

Sarah Rastegar, Hazel Doughty, and Cees Snoek. Learn to categorize or categorize to learn? self-coding for generalized category discovery. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 72794–72818. Curran Associates, Inc., 2023. doi: 10.52202/075280-3183. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/e6789e468c65a7816760a00a487d3c4e-Paper-Conference.pdf.

Sarah Rastegar, Mohammadreza Salehi, Yuki M. Asano, Hazel Doughty, and Cees G. M. Snoek. Selex: Self-expertise in fine-grained generalized category discovery, 2024. URL https://arxiv.org/abs/2408.14371.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vision*, 115(3):211–252, December 2015. ISSN 0920-5691. doi: 10.1007/s11263-015-0816-y. URL https://doi.org/10.1007/s11263-015-0816-y.

M. Saquib Sarfraz, Vivek Sharma, and Rainer Stiefelhagen. Efficient parameter-free clustering using first neighbor relations, 2019. URL https://arxiv.org/abs/1902.11266.

Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Generalized category discovery, 2022a. URL https://arxiv.org/abs/2201.02609.

Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Open-set recognition: a good closed-set classifier is all you need?, 2022b. URL https://arxiv.org/abs/2110.06207.

Ulrike von Luxburg. A tutorial on spectral clustering, 2007. URL https://arxiv.org/abs/0711.0189.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

Hongjun Wang, Sagar Vaze, and Kai Han. SPTNet: An efficient alternative framework for generalized category discovery with spatial prompt tuning. In *International Conference on Learning Representations*, volume 2024, pp. 53803–53838, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/eccc6e11878857e87ec7dd109eaa9eeb-Paper-Conference.pdf.

Hongjun Wang, Sagar Vaze, and Kai Han. Hilo: A learning framework for generalized category discovery robust to domain shifts. In *International Conference on Learning Representations (ICLR)*, 2025.

Xin Wen, Bingchen Zhao, and Xiaojuan Qi. Parametric classification for generalized category discovery: A baseline study, 2023. URL https://arxiv.org/abs/2211.11727.

Lihi Zelnik-manor and Pietro Perona. Self-tuning spectral clustering. In L. Saul, Y. Weiss, and L. Bottou (eds.), *Advances in Neural Information Processing Systems*, volume 17. MIT Press, 2004. URL https://proceedings.neurips.cc/paper_files/paper/2004/file/40173ea48d9567f1f393b20c855bb40b-Paper.pdf.

Sheng Zhang, Salman Khan, Zhiqiang Shen, Muzammal Naseer, Guangyi Chen, and Fahad Khan. Promptcal: Contrastive affinity learning via auxiliary prompts for generalized novel category discovery, 2023. URL https://arxiv.org/abs/2212.05590.

Bingchen Zhao, Xin Wen, and Kai Han. Learning semi-supervised gaussian mixture models for generalized category discovery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16623–16633, October 2023.

Zhun Zhong, Enrico Fini, Subhankar Roy, Zhiming Luo, Elisa Ricci, and Nicu Sebe. Neighborhood contrastive learning for novel class discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10867–10875, June 2021.

Dengyong Zhou, Olivier Bousquet, Thomas Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In S. Thrun, L. Saul, and B. Schölkopf (eds.), *Advances in Neural Information Processing Systems*, volume 16. MIT Press, 2003. URL https://proceedings.neurips.cc/paper_files/paper/2003/file/87682805257e619d49b8e0dfdc14affa-Paper.pdf.

# APPENDIX


**A Additional Analysis** **15**

**B Threshold Diagnostics** **17**

**C Complete Numerical Results** **19**

**D Proof of Proposition 1** **24**

**E Anchor-selection threshold sensitivity** **25**

**F Paired variability on the additional datasets** **26**

**G Computational overhead** **27**

**H Transferability to an alternative GCD predictor** **28**

# A Additional Analysis

We examine class-wise sensitivity, paired uncertainty, and correction outcomes before presenting threshold diagnostics in Appendix B and complete numerical results in Appendix C.

## A.1 Class-wise sensitivity to neighborhood size

Figure 6 extends the overall comparison in Figure 5 to Old and New classes on the 5,794-image CUB development evaluation. Both configurations use maximum aggregation; Full uses complete-density selection with cap 20, and Budget uses budget normalization without a cap. These curves compare the two complete configurations. Table 2 separately isolates normalization and the cap.

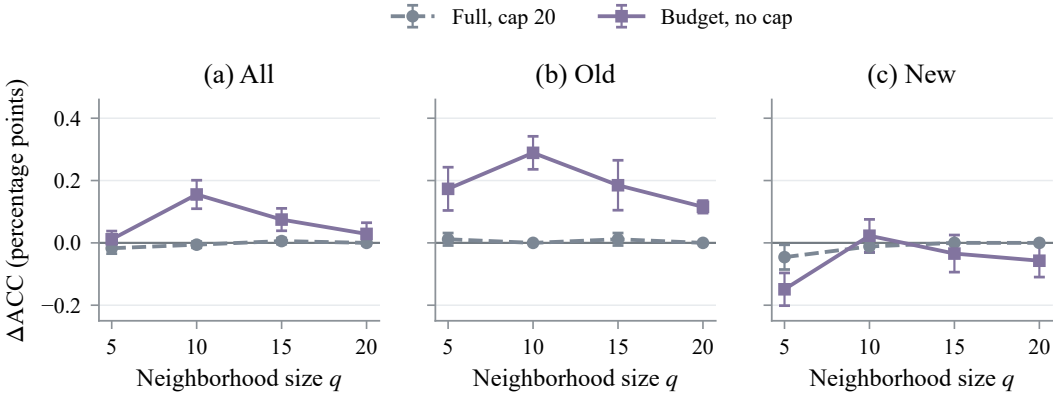



Figure 6: Class-wise sensitivity to $q$. Points and error bars show the mean and standard deviation of paired accuracy changes across three clustering runs, relative to each run's uncorrected predictions. Budget selection improves mean All and Old accuracy at all four values; mean New accuracy improves at $q = 10$ and decreases at $q = 5, 15, 20$.

The class-wise curves separate overall gains from their composition. Old accuracy increases throughout the evaluated range, while the New-class response changes sign. The strongest mean All gain occurs at $q = 10$, where the mean New change is also positive. At $q = 5$, one run has negative net correction despite a positive mean All change. Table 7 retains all individual outcomes.

## A.2 Paired uncertainty across correction configurations

Figure 7 compares each configuration with its uncorrected predictions under the fixed Hungarian mapping. For the image bootstrap, we first average each image's paired correctness differences across clustering runs and then resample images. The resulting intervals quantify image-level uncertainty in the mean paired accuracy change.

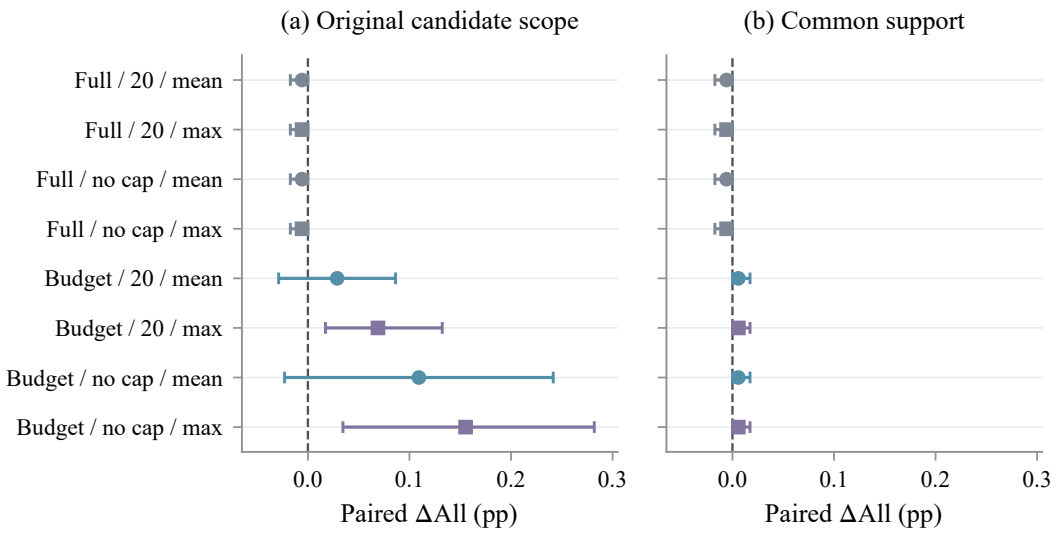



Figure 7: Paired All accuracy changes and 95% image-bootstrap intervals on the 5,794-image CUB evaluation. Labels specify normalization, size cap, and aggregation. The vertical line marks zero change. The two panels distinguish the original candidate scope from common support.

In the original candidate scope, budget normalization with maximum aggregation yields positive lower interval bounds with and without cap 20. The mean-aggregation intervals cross zero. Under common support, all four budget configurations have the same small change and an interval touching zero. Together with the paired support analysis in Section 4.3, this contrast connects the larger gains to the additional queries reached by the selected anchors.

### A.3 Correction outcomes across runs

Figure 8 decomposes the changes at $q = 10$. Across the three runs, uncapped budget selection produces 39 fixes and 20 harms among 87 modifications with mean aggregation, and 43 fixes and 16 harms among 85 modifications with maximum aggregation. Both variants yield positive net correction in every run. Wrong-to-wrong changes explain why modification counts alone do not measure improvement.

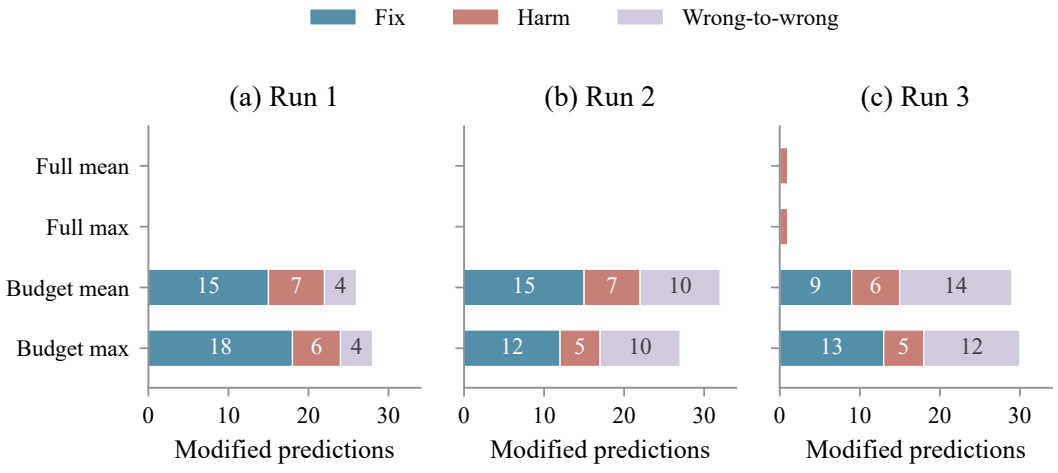



Figure 8: Correction outcomes on the 5,794-image CUB evaluation at $q = 10$. Full uses cap 20 and Budget has no size cap. Stacked counts distinguish fixes, harms, and wrong-to-wrong changes; net correction is fixes minus harms. Each panel shows one clustering run.

## B THRESHOLD DIAGNOSTICS

These post-hoc diagnostics use the separate 4,496-image CUB evaluation underlying Table 3. Coverage is the fraction of all evaluated seed–image events whose predictions change, and precision is the fraction of those changes that fix an error. The markers in Figures 9 and 10 indicate the threshold settings used to obtain the results in Table 3.

### B.1 CORRECTION QUALITY AS COVERAGE CHANGES

Figure 9 separates correction precision from net correction and compares the original candidate scope with common support. At the reported operating points in the original scope, the size-aware variants make fewer changes and incur fewer harms than KNN and nearest-member correction. Restricting the latter two methods to common support changes their net corrections from $-4$ and $-3$ to $+7$ and $+12$, respectively. Candidate scope therefore contributes to the observed differences alongside the scoring rule.

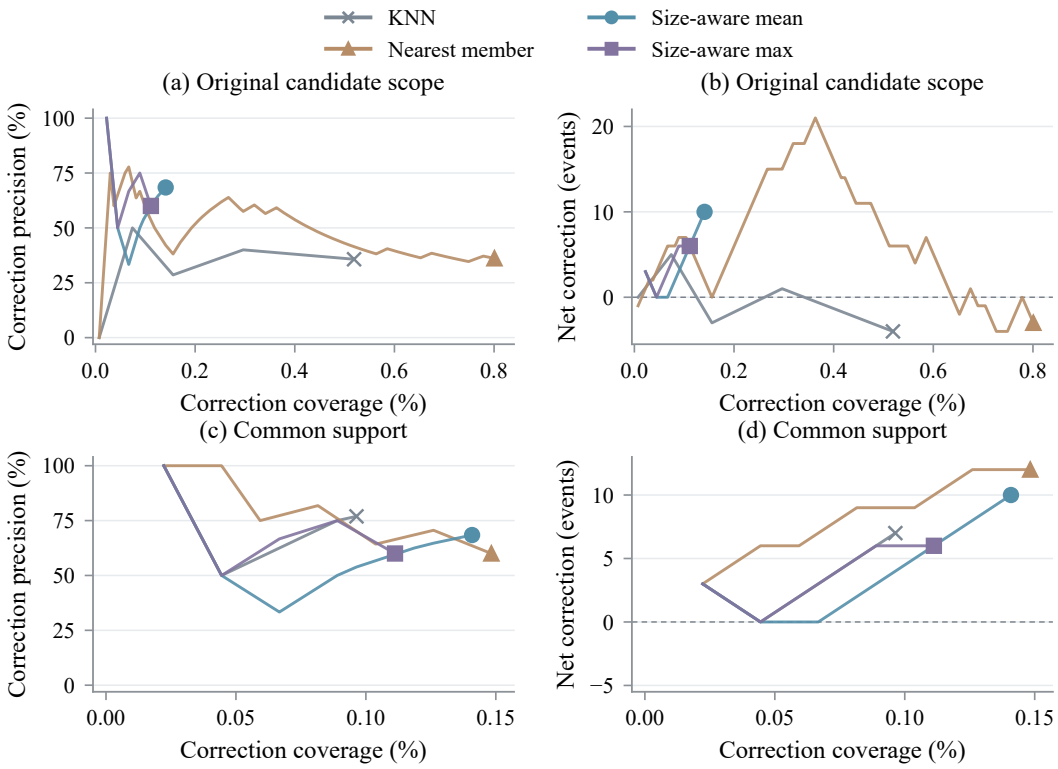



Figure 9: Correction quality versus coverage on the 4,496-image CUB evaluation. Lines trace the recorded threshold sweep; markers identify the reported operating points. The upper and lower rows use the original candidate scope and common support, respectively. Net correction counts fixes minus harms across three clustering runs.

### B.2 THRESHOLDS AND CORRECTION COVERAGE

Figure 10 shows how increasingly restrictive thresholds reduce coverage. KNN thresholds count neighbor votes for the second candidate. The other panels threshold the support for the second-nearest-center category minus that for the nearest-center category (rank 2 minus rank 1). Each method consequently retains its own threshold axis. For the size-aware variants, the original-scope and common-support curves coincide in these records. The sweeps describe the recorded scores; the reported operating points are unchanged.

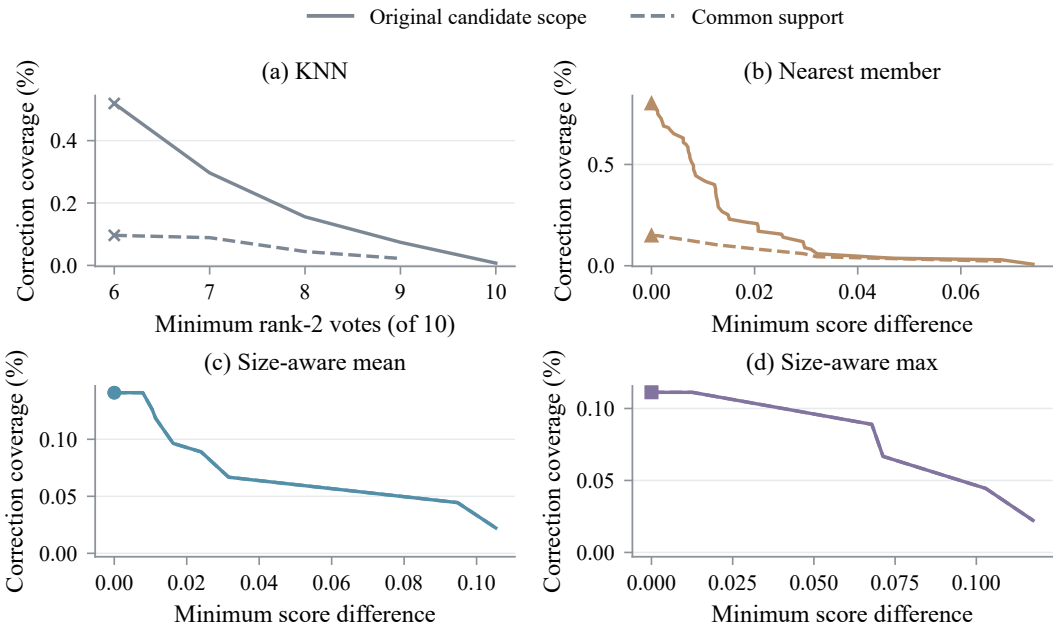



Figure 10: Threshold–coverage relations for the four correction rules. Solid and dashed curves denote the original candidate scope and common support; coincident curves overlap. Markers indicate the reported thresholds. Threshold units differ across methods, while coverage uses the same event denominator.

# C Complete Numerical Results

The following tables retain the individual results and their original numerical precision. Full denotes complete-graph normalization and Budget denotes connection-budget normalization. A cap of $\infty$ means no explicit size cap. In correction statistics, $M$, $F$, and $H$ count modified predictions, fixes, and harms, respectively; $\mathrm{Net} = F - H$, and the remaining $M - F - H$ changes are wrong-to-wrong. Rows for our budget-normalized configurations are shaded blue.

## C.1 Aggregate accuracy and paired intervals

Tables 4 and 5 report dispersion and paired intervals separately. All entries use the fixed Hungarian mapping obtained from the uncorrected predictions.

Table 4: CUB accuracy (%) as mean ± standard deviation across three clustering runs on 5,794 images.

| Normalization | Cap | Aggregation | All | Old | New |
|---|---|---|---|---|---|
| *Original scope* | | | | | |
| Uncorrected | – | – | 84.4264±0.6582 | 85.4716±1.3285 | 83.3906±0.8559 |
| Full | 20 | mean | 84.4207±0.6486 | 85.4716±1.3285 | 83.3792±0.8448 |
| Full | 20 | max | 84.4207±0.6486 | 85.4716±1.3285 | 83.3792±0.8448 |
| Full | $\infty$ | mean | 84.4207±0.6486 | 85.4716±1.3285 | 83.3792±0.8448 |
| Full | $\infty$ | max | 84.4207±0.6486 | 85.4716±1.3285 | 83.3792±0.8448 |
| Budget | 20 | mean | 84.4552±0.6319 | 85.5525±1.2770 | 83.3677±0.8361 |
| Budget | 20 | max | 84.4955±0.6463 | 85.5987±1.3169 | 83.4021±0.8361 |
| Budget | $\infty$ | mean | 84.5357±0.6105 | 85.8414±1.2854 | 83.2417±0.8142 |
| Budget | $\infty$ | max | 84.5818±0.6573 | 85.7605±1.3341 | 83.4135±0.8200 |
| *Common support* | | | | | |
| Uncorrected | – | – | 84.4264±0.6582 | 85.4716±1.3285 | 83.3906±0.8559 |
| Full | 20 | mean | 84.4207±0.6486 | 85.4716±1.3285 | 83.3792±0.8448 |
| Full | 20 | max | 84.4207±0.6486 | 85.4716±1.3285 | 83.3792±0.8448 |
| Full | $\infty$ | mean | 84.4207±0.6486 | 85.4716±1.3285 | 83.3792±0.8448 |
| Full | $\infty$ | max | 84.4207±0.6486 | 85.4716±1.3285 | 83.3792±0.8448 |
| Budget | 20 | mean | 84.4322±0.6559 | 85.4716±1.3285 | 83.4021±0.8361 |
| Budget | 20 | max | 84.4322±0.6559 | 85.4716±1.3285 | 83.4021±0.8361 |
| Budget | $\infty$ | mean | 84.4322±0.6559 | 85.4716±1.3285 | 83.4021±0.8361 |
| Budget | $\infty$ | max | 84.4322±0.6559 | 85.4716±1.3285 | 83.4021±0.8361 |

Table 5: Paired All accuracy changes and 95% image-bootstrap confidence intervals (CI), in percentage points.

| Normalization | Cap | Aggregation | ΔAll | 95% CI |
|---|---|---|---|---|
| *Original scope* | | | | |
| Uncorrected | – | – | +0.0000 | [+0.0000,+0.0000] |
| Full | 20 | mean | -0.0058 | [-0.0173,+0.0000] |
| Full | 20 | max | -0.0058 | [-0.0173,+0.0000] |
| Full | $\infty$ | mean | -0.0058 | [-0.0173,+0.0000] |
| Full | $\infty$ | max | -0.0058 | [-0.0173,+0.0000] |
| Budget | 20 | mean | +0.0288 | [-0.0288,+0.0863] |
| Budget | 20 | max | +0.0690 | [+0.0173,+0.1323] |
| Budget | $\infty$ | mean | +0.1093 | [-0.0230,+0.2416] |
| Budget | $\infty$ | max | +0.1553 | [+0.0345,+0.2820] |
| *Common support* | | | | |
| Uncorrected | – | – | +0.0000 | [+0.0000,+0.0000] |
| Full | 20 | mean | -0.0058 | [-0.0173,+0.0000] |
| Full | 20 | max | -0.0058 | [-0.0173,+0.0000] |
| Full | $\infty$ | mean | -0.0058 | [-0.0173,+0.0000] |
| Full | $\infty$ | max | -0.0058 | [-0.0173,+0.0000] |
| Budget | 20 | mean | +0.0058 | [+0.0000,+0.0173] |
| Budget | 20 | max | +0.0058 | [+0.0000,+0.0173] |
| Budget | $\infty$ | mean | +0.0058 | [+0.0000,+0.0173] |
| Budget | $\infty$ | max | +0.0058 | [+0.0000,+0.0173] |

KNN yields All $84.6220 \pm 0.6070$, Old $86.4540 \pm 1.3914$, and New $82.8064 \pm 0.7936$ on this evaluation.

### C.2 INDIVIDUAL CUB RESULTS

**Initial assignments and correction eligibility.** The cached clustering assignments can differ from the nearest returned centers. Among low-margin queries, requiring the initial assignment to equal the nearest-center category ($b_i = k_{i,1}$; Section 3.4) excludes 7, 101, and 0 events in runs 1–3, leaving 382, 360, and 391 eligible queries. The second clustering output contains one empty cluster with a non-finite center; this center appears in neither the original assignments nor the cached top-two rankings. All paired variants use the same cached assignments and centers.
Table 6 joins accuracy and correction counts for each run. Full uses cap 20; Budget has no explicit cap.

Table 6: Individual CUB results on 5,794 images at $q = 10$. Accuracy columns are percentages; correction columns are event counts.

| Method | Aggregation | All | Old | New | $M$ | $F$ | $H$ | Net |
|---|---|---|---|---|---|---|---|---|
| *Run 1* | | | | | | | | |
| Uncorrected | – | 84.2423 | 86.0957 | 82.4055 | 0 | 0 | 0 | 0 |
| Full | mean | 84.2423 | 86.0957 | 82.4055 | 0 | 0 | 0 | 0 |
| Full | max | 84.2423 | 86.0957 | 82.4055 | 0 | 0 | 0 | 0 |
| Budget | mean | 84.3804 | 86.4771 | 82.3024 | 26 | 15 | 7 | 8 |
| Budget | max | 84.4494 | 86.4424 | 82.4742 | 28 | 18 | 6 | 12 |
| KNN | – | 84.5530 | 87.2399 | 81.8900 | 108 | 52 | 34 | 18 |
| *Run 2* | | | | | | | | |
| Uncorrected | – | 83.8799 | 83.9459 | 83.8144 | 0 | 0 | 0 | 0 |
| Full | mean | 83.8799 | 83.9459 | 83.8144 | 0 | 0 | 0 | 0 |
| Full | max | 83.8799 | 83.9459 | 83.8144 | 0 | 0 | 0 | 0 |
| Budget | mean | 84.0179 | 84.3620 | 83.6770 | 32 | 15 | 7 | 8 |
| Budget | max | 84.0007 | 84.2233 | 83.7801 | 27 | 12 | 5 | 7 |
| KNN | – | 84.0525 | 84.8474 | 83.2646 | 117 | 48 | 38 | 10 |
| *Run 3* | | | | | | | | |
| Uncorrected | – | 85.1571 | 86.3731 | 83.9519 | 0 | 0 | 0 | 0 |
| Full | mean | 85.1398 | 86.3731 | 83.9175 | 1 | 0 | 1 | -1 |
| Full | max | 85.1398 | 86.3731 | 83.9175 | 1 | 0 | 1 | -1 |
| Budget | mean | 85.2088 | 86.6852 | 83.7457 | 29 | 9 | 6 | 3 |
| Budget | max | 85.2951 | 86.6158 | 83.9863 | 30 | 13 | 5 | 8 |
| KNN | – | 85.2606 | 87.2746 | 83.2646 | 114 | 46 | 40 | 6 |

### C.3 NEIGHBORHOOD-SIZE RESULTS

Table 7 contains the individual results underlying Figure 6. Both selectors use maximum aggregation. Full uses cap 20 and Budget has no size cap. Supported counts queries satisfying $m_i \leq \tau_m$ and $b_i = k_{i,1}$ that have anchors for both candidate classes after excluding all groups containing the query. At $q = 10$, the counts sum to 5 Full and 200 Budget seed–image events, corresponding to 3 and 105 distinct images, respectively.

Table 7: Neighborhood-size sensitivity on 5,794 CUB images. Sup. denotes the number of queries with two-sided anchor support.

| Run | Selector | All | Old | New | Sup. | $M$ | $F$ | $H$ | Net |
|---|---|---|---|---|---|---|---|---|---|
| $q = 5$ | | | | | | | | | |
| 1 | Full | 84.2251 | 86.1304 | 82.3368 | 8 | 3 | 1 | 2 | -1 |
| 1 | Budget | 84.2768 | 86.2691 | 82.3024 | 60 | 28 | 13 | 11 | 2 |
| 2 | Full | 83.8799 | 83.9459 | 83.8144 | 5 | 1 | 0 | 0 | 0 |
| 2 | Budget | 83.8971 | 84.1886 | 83.6082 | 56 | 28 | 9 | 8 | 1 |
| 3 | Full | 85.1225 | 86.3731 | 83.8832 | 3 | 2 | 0 | 2 | -2 |
| 3 | Budget | 85.1398 | 86.4771 | 83.8144 | 48 | 22 | 8 | 9 | -1 |
| $q = 10$ | | | | | | | | | |
| 1 | Full | 84.2423 | 86.0957 | 82.4055 | 2 | 0 | 0 | 0 | 0 |
| 1 | Budget | 84.4494 | 86.4424 | 82.4742 | 67 | 28 | 18 | 6 | 12 |
| 2 | Full | 83.8799 | 83.9459 | 83.8144 | 1 | 0 | 0 | 0 | 0 |
| 2 | Budget | 84.0007 | 84.2233 | 83.7801 | 57 | 27 | 12 | 5 | 7 |
| 3 | Full | 85.1398 | 86.3731 | 83.9175 | 2 | 1 | 0 | 1 | -1 |
| 3 | Budget | 85.2951 | 86.6158 | 83.9863 | 76 | 30 | 13 | 5 | 8 |
| $q = 15$ | | | | | | | | | |
| 1 | Full | 84.2423 | 86.0957 | 82.4055 | 0 | 0 | 0 | 0 | 0 |
| 1 | Budget | 84.3286 | 86.2344 | 82.4399 | 48 | 22 | 11 | 6 | 5 |
| 2 | Full | 83.8971 | 83.9806 | 83.8144 | 2 | 1 | 1 | 0 | 1 |
| 2 | Budget | 83.9834 | 84.2233 | 83.7457 | 43 | 21 | 11 | 5 | 6 |
| 3 | Full | 85.1571 | 86.3731 | 83.9519 | 1 | 0 | 0 | 0 | 0 |
| 3 | Budget | 85.1916 | 86.5118 | 83.8832 | 56 | 27 | 9 | 7 | 2 |
| $q = 20$ | | | | | | | | | |
| 1 | Full | 84.2423 | 86.0957 | 82.4055 | 0 | 0 | 0 | 0 | 0 |
| 1 | Budget | 84.2596 | 86.1997 | 82.3368 | 32 | 14 | 6 | 5 | 1 |
| 2 | Full | 83.8799 | 83.9459 | 83.8144 | 0 | 0 | 0 | 0 | 0 |
| 2 | Budget | 83.9489 | 84.0846 | 83.8144 | 31 | 13 | 7 | 3 | 4 |
| 3 | Full | 85.1571 | 86.3731 | 83.9519 | 0 | 0 | 0 | 0 | 0 |
| 3 | Budget | 85.1571 | 86.4771 | 83.8488 | 41 | 17 | 5 | 5 | 0 |

### C.4 FACTORIZED NORMALIZATION AND CAP ABLATION

Table 8 gives the complete per-run ablation behind Table 2. Normalization, cap, and aggregation appear in separate columns to expose the paired comparisons.

Table 8: Factorized CUB ablation on 5,794 images at $q = 10$, retaining all eight configurations in each run.

| Norm. | Cap | Agg. | All | Old | New | $M$ | $F$ | $H$ | Net |
|---|---|---|---|---|---|---|---|---|---|
| *Run 1* | | | | | | | | | |
| Full | 20 | mean | 84.2423 | 86.0957 | 82.4055 | 0 | 0 | 0 | 0 |
| Full | 20 | max | 84.2423 | 86.0957 | 82.4055 | 0 | 0 | 0 | 0 |
| Full | $\infty$ | mean | 84.2423 | 86.0957 | 82.4055 | 0 | 0 | 0 | 0 |
| Full | $\infty$ | max | 84.2423 | 86.0957 | 82.4055 | 0 | 0 | 0 | 0 |
| Budget | 20 | mean | 84.2768 | 86.1650 | 82.4055 | 4 | 3 | 1 | 2 |
| Budget | 20 | max | 84.3286 | 86.2344 | 82.4399 | 5 | 5 | 0 | 5 |
| Budget | $\infty$ | mean | 84.3804 | 86.4771 | 82.3024 | 26 | 15 | 7 | 8 |
| Budget | $\infty$ | max | 84.4494 | 86.4424 | 82.4742 | 28 | 18 | 6 | 12 |
| *Run 2* | | | | | | | | | |
| Full | 20 | mean | 83.8799 | 83.9459 | 83.8144 | 0 | 0 | 0 | 0 |
| Full | 20 | max | 83.8799 | 83.9459 | 83.8144 | 0 | 0 | 0 | 0 |
| Full | $\infty$ | mean | 83.8799 | 83.9459 | 83.8144 | 0 | 0 | 0 | 0 |
| Full | $\infty$ | max | 83.8799 | 83.9459 | 83.8144 | 0 | 0 | 0 | 0 |
| Budget | 20 | mean | 83.9317 | 84.0846 | 83.7801 | 14 | 5 | 2 | 3 |
| Budget | 20 | max | 83.9489 | 84.0846 | 83.8144 | 13 | 5 | 1 | 4 |
| Budget | $\infty$ | mean | 84.0179 | 84.3620 | 83.6770 | 32 | 15 | 7 | 8 |
| Budget | $\infty$ | max | 84.0007 | 84.2233 | 83.7801 | 27 | 12 | 5 | 7 |
| *Run 3* | | | | | | | | | |
| Full | 20 | mean | 85.1398 | 86.3731 | 83.9175 | 1 | 0 | 1 | -1 |
| Full | 20 | max | 85.1398 | 86.3731 | 83.9175 | 1 | 0 | 1 | -1 |
| Full | $\infty$ | mean | 85.1398 | 86.3731 | 83.9175 | 1 | 0 | 1 | -1 |
| Full | $\infty$ | max | 85.1398 | 86.3731 | 83.9175 | 1 | 0 | 1 | -1 |
| Budget | 20 | mean | 85.1571 | 86.4078 | 83.9175 | 2 | 1 | 1 | 0 |
| Budget | 20 | max | 85.2088 | 86.4771 | 83.9519 | 3 | 3 | 0 | 3 |
| Budget | $\infty$ | mean | 85.2088 | 86.6852 | 83.7457 | 29 | 9 | 6 | 3 |
| Budget | $\infty$ | max | 85.2951 | 86.6158 | 83.9863 | 30 | 13 | 5 | 8 |

The unchanged predictions between Full/20 and Full/$\infty$ are not caused by the size cap itself. On the 5,794-image CUB development evaluation across three clustering seeds, these two configurations produce identical anchors, supported queries, and correction scores. The largest retained anchor group under complete-graph normalization contains 16 members, and all larger candidate groups fail the density threshold under the evaluated setting. Therefore, removing the explicit cap does not expand the candidate set for complete-density selection.

### C.5 CIFAR-100 CORRECTION RESULTS

These results use a CIFAR-100-trained encoder and the 30,000 unlabeled training images. The size-aware variants use $q = 10$, $\tau_m = 0.2$, and the same confirmation rule as in the main experiments. Table 9 reports correction counts and rates. Fix rate divides fixes by the number of initially incorrect eligible queries; damage rate divides harms by the number of initially correct eligible queries. Both are percentages.

Table 9: CIFAR-100 correction statistics. Counts and recorded rates (%) are reported for each clustering run.

| Method | Agg. | $M$ | $F$ | $H$ | Net | Fix rate | Damage rate |
|---|---|---|---|---|---|---|---|
| *Run 1* | | | | | | | |
| Uncorrected | – | 0 | 0 | 0 | +0 | 0.0000 | 0.0000 |
| KNN | – | 132 | 74 | 41 | +33 | 13.6531 | 7.7358 |
| Complete density | mean | 70 | 20 | 42 | -22 | 3.6900 | 7.9245 |
| Complete density | max | 86 | 30 | 46 | -16 | 5.5351 | 8.6792 |
| Size-aware | mean | 14 | 8 | 5 | +3 | 1.4760 | 0.9434 |
| Size-aware | max | 12 | 8 | 3 | +5 | 1.4760 | 0.5660 |
| *Run 2* | | | | | | | |
| Uncorrected | – | 0 | 0 | 0 | +0 | 0.0000 | 0.0000 |
| KNN | – | 132 | 74 | 41 | +33 | 13.6531 | 7.7358 |
| Complete density | mean | 70 | 20 | 42 | -22 | 3.6900 | 7.9245 |
| Complete density | max | 86 | 30 | 46 | -16 | 5.5351 | 8.6792 |
| Size-aware | mean | 14 | 8 | 5 | +3 | 1.4760 | 0.9434 |
| Size-aware | max | 12 | 8 | 3 | +5 | 1.4760 | 0.5660 |
| *Run 3* | | | | | | | |
| Uncorrected | – | 0 | 0 | 0 | +0 | 0.0000 | 0.0000 |
| KNN | – | 131 | 75 | 39 | +36 | 13.7868 | 7.3585 |
| Complete density | mean | 70 | 20 | 42 | -22 | 3.6765 | 7.9245 |
| Complete density | max | 86 | 30 | 46 | -16 | 5.5147 | 8.6792 |
| Size-aware | mean | 14 | 8 | 5 | +3 | 1.4706 | 0.9434 |
| Size-aware | max | 12 | 8 | 3 | +5 | 1.4706 | 0.5660 |
| *Run 4* | | | | | | | |
| Uncorrected | – | 0 | 0 | 0 | +0 | 0.0000 | 0.0000 |
| KNN | – | 131 | 75 | 39 | +36 | 13.7868 | 7.3585 |
| Complete density | mean | 70 | 20 | 42 | -22 | 3.6765 | 7.9245 |
| Complete density | max | 86 | 30 | 46 | -16 | 5.5147 | 8.6792 |
| Size-aware | mean | 14 | 8 | 5 | +3 | 1.4706 | 0.9434 |
| Size-aware | max | 12 | 8 | 3 | +5 | 1.4706 | 0.5660 |
| *Run 5* | | | | | | | |
| Uncorrected | – | 0 | 0 | 0 | +0 | 0.0000 | 0.0000 |
| KNN | – | 131 | 75 | 39 | +36 | 13.7868 | 7.3585 |
| Complete density | mean | 70 | 20 | 42 | -22 | 3.6765 | 7.9245 |
| Complete density | max | 86 | 30 | 46 | -16 | 5.5147 | 8.6792 |
| Size-aware | mean | 14 | 8 | 5 | +3 | 1.4706 | 0.9434 |
| Size-aware | max | 12 | 8 | 3 | +5 | 1.4706 | 0.5660 |

## D PROOF OF PROPOSITION 1

We restate the implicit size ceiling from Section 3.2 and provide its proof.

**Proposition 1 (Implicit size ceiling, restated).** In a mutual-$q$NN graph, a complete-density threshold $\tau_\rho \in (0, 1]$ cannot be satisfied by any group with $n > 1 + q/\tau_\rho$, even if every vertex exhausts its internal degree capacity.

*Proof.* For a group $G$ of size $n \geq 2$ in view $v$, recall that $E_v(G) = \sum_{i,j \in G} A_{ij}^{(v)}$ counts each undirected edge twice. Each vertex has at most $\min(q, n-1)$ neighbors within $G$, so $E_v(G) \leq n \min(q, n-1) \leq nq$. Hence $\rho_v^{\text{full}}(G) = E_v(G)/[n(n-1)] \leq q/(n-1)$, and $\rho_v^{\text{full}}(G) \geq \tau_\rho$ implies $n \leq 1 + q/\tau_\rho$.

## E ANCHOR-SELECTION THRESHOLD SENSITIVITY

We vary one anchor-selection threshold at a time on the 5,794-image CUB development split, keeping $n_{\min} = 3$, $q = 10$, $\tau_m = 0.2$, and the other selection thresholds at their defaults. The analysis uses the same fixed features, uncorrected predictions, and core anchor-correction protocol as Figure 5, before additional neighborhood confirmation, with budget normalization and no size cap. All entries average the same clustering runs; the Hungarian mapping is fixed within each run.

Table 10: One-at-a-time sensitivity of anchor selection. Anchors counts retained groups; Supported counts eligible queries with two-sided anchor support after self-group exclusion. Paired ΔAll is in percentage points. Shaded rows indicate the default threshold settings.

| Parameter | Value | Anchors | Supported | ΔAll (mean) | ΔAll (max) |
|---|---|---|---|---|---|
| $\tau_J$ | 0.7 | 180.67 | 75.67 | +0.1036 | +0.1381 |
| | 0.8 | 165.67 | 66.67 | +0.1093 | +0.1553 |
| | 0.9 | 123.33 | 39.00 | +0.0403 | +0.0575 |
| $\tau_Q$ | 0.7 | 182.33 | 105.00 | +0.1668 | +0.2301 |
| | 0.8 | 165.67 | 66.67 | +0.1093 | +0.1553 |
| | 0.9 | 68.00 | 12.33 | -0.0288 | 0.0000 |
| $\tau_\rho$ | 0.4 | 169.33 | 76.00 | +0.1151 | +0.1323 |
| | 0.5 | 165.67 | 66.67 | +0.1093 | +0.1553 |
| | 0.6 | 129.33 | 39.00 | +0.0460 | +0.1208 |

Tighter thresholds reduce anchor retention and two-sided support. Both aggregation rules retain positive average gains across the tested $\tau_J$ and $\tau_\rho$ ranges. The stricter $\tau_Q = 0.9$ leaves 12.33 supported queries per run and yields $-0.0288$ and $0.0000$ pp for mean and maximum aggregation, respectively. The default setting remains $(\tau_J, \tau_Q, \tau_\rho) = (0.8, 0.8, 0.5)$.

## F PAIRED VARIABILITY ON THE ADDITIONAL DATASETS

For the five additional datasets in Table 1, we compute the All-accuracy difference between corrected and uncorrected predictions within each clustering seed, then report its mean and sample standard deviation across seeds 1–5 (denominator $5 - 1$). The checkpoint, training seed, and previously reported correction configuration remain fixed, with $q = 10$ and $\tau_m = 0.2$. Thus, the SD measures variation in the paired correction effect across clustering initializations, rather than variation across independently trained models.

Table 11: Paired $\Delta$All (mean $\pm$ sample SD, percentage points) on the five additional datasets. Green and red indicate positive and negative mean changes, respectively.

| Dataset | Mean aggregation | Max aggregation |
|---|---|---|
| Cars | $-0.0130 \pm 0.0313$ | $-0.0130 \pm 0.0313$ |
| Aircraft | $-0.0240 \pm 0.0089$ | $-0.0240 \pm 0.0089$ |
| CIFAR-10 | $+0.2896 \pm 0.0015$ | $+0.2896 \pm 0.0015$ |
| CIFAR-100 | $+0.0800 \pm 0.0000$ | $+0.0800 \pm 0.0000$ |
| ImageNet-100 | $+0.3872 \pm 0.1928$ | $+0.3872 \pm 0.1928$ |

Cars has a mean change smaller in magnitude than its across-seed SD. Aircraft has a negative paired change in all five runs. On CIFAR-100, the paired accuracy gain is $+0.0800$ percentage points in all five clustering runs, giving a standard deviation of zero. CIFAR-10 and ImageNet-100 have positive paired changes in all five runs, with greater across-seed variation on ImageNet-100.

## G COMPUTATIONAL OVERHEAD

SAAS operates on cached features and uncorrected predictions without retraining. Table 12 reports its full post-processing cost, including top-two center assignment and margin computation, graph construction, anchor selection, support computation, cross-view neighborhood confirmation, and final correction. Feature extraction, clustering, input loading, and output writing are outside this timed interval.

Table 12: SAAS post-processing cost. Time is the mean $\pm$ sample SD of three independent timing runs per aggregation rule, with the cached inputs and clustering seed fixed. Memory is the maximum additional usage across both aggregation rules and all timing runs; GB denotes $10^9$ bytes.

| Dataset | # unlabeled | Post-processing time (s) | | Peak additional memory (GB) | |
|---|---|---|---|---|---|
| | images | Mean | Max | RAM | GPU |
| CUB | 4,496 | $4.02 \pm 2.01$ | $5.70 \pm 2.48$ | 0.337 | 0.033 |
| CIFAR-100 | 30,000 | $52.74 \pm 0.04$ | $55.14 \pm 3.91$ | 0.854 | 0.164 |

We measure post-processing cost on an Intel Xeon Platinum 8352Y CPU (2.20 GHz, two CPU/BLAS threads) and an NVIDIA A800 80 GB GPU. Each run uses a fresh process, with input arrays loaded and the CUDA context initialized before timing. Wall-clock time covers the complete post-processing pass, with GPU synchronization at the timing boundaries. Mutual graphs with $q = 10$ are built on the CPU, and confirmation neighborhoods are constructed on the GPU; both are rebuilt for each run.

Memory usage is reported relative to the level immediately before post-processing, when the inputs are already resident. Additional RAM is the peak increase in process resident memory, sampled every 2 ms and at stage boundaries. GPU memory is the peak increase in CUDA-allocated memory. Peak additional CUDA-reserved memory is 0.038 GB on CUB and 0.178 GB on CIFAR-100.

## H TRANSFERABILITY TO AN ALTERNATIVE GCD PREDICTOR

To examine whether SAAS depends on the Hyp-SelEx prediction source, we apply the same anchor selection and correction procedure to PromptCAL predictions. All SAAS configurations remain unchanged.

Table 13: Transferability of SAAS to PromptCAL predictions. All and SAAS All report accuracy (%) before and after correction; ΔAll is the paired change in percentage points.

| Predictor | All | SAAS All | ΔAll |
|---|---|---|---|
| PromptCAL | 62.4377 | 62.4911 | $+0.0534 \pm 0.0122$ |

Across five clustering runs, SAAS consistently improves the baseline prediction. Both mean and maximum aggregation variants produce identical corrected predictions, with 14 fixes and 2 harms in each run.