DD-FSC: DIRICHLET PROCESS-GUIDED DIFFUSION MODEL FOR FEW-SHOT CLUSTERING

Anonymous authors
Paper under double-blind review

ABSTRACT

In few-shot deep clustering, partial category coverage and an unknown number of latent categories make it difficult to propagate sparse semantic supervision while reliably recovering the complete clustering structure. To address this challenge, we propose DD-FSC, a nonparametric few-shot clustering framework that jointly models feature-space density and latent cluster structure. DD-FSC characterizes the feature distribution with a diffusion model and employs score-guided Langevin dynamics to propagate limited supervision toward high-density semantic regions, thereby identifying high-confidence same-category samples and constructing density-aware positive pairs. Meanwhile, Dirichlet process (DP) inference models the evolving cluster structure and adaptively determines the effective number of clusters without requiring it to be specified in advance. Theoretically, we show that the deterministic drift of score-guided Langevin dynamics is positively aligned with the modeled log-density gradient, providing a principled basis for density-guided semantic propagation. Through this density-guided semantic expansion and adaptive structure inference, DD-FSC extends sparse supervision beyond the observed categories while recovering the underlying clustering structure. Extensive experiments demonstrate superior clustering performance under both known- and unknown-cluster-number settings, together with accurate estimation of the latent category count.

1 INTRODUCTION

Deep clustering groups unlabeled samples into semantic categories by jointly learning representations and cluster assignments (Xie et al., 2016; Li et al., 2021). In practical settings, a small number of labeled samples may be available, but they often cover only part of the underlying category space (Cao et al., 2022; Rizve et al., 2022). The remaining unlabeled data therefore contain both observed and novel categories (Vaze et al., 2022; Li et al., 2025), raising a central question: how can sparse and partially covered supervision be reliably propagated to recover the complete latent clustering structure? This challenge is further compounded when the number of categories K is unknown and must be inferred during learning (Ronen et al., 2022; Luo et al., 2025).

Existing approaches address this setting mainly by improving pseudo-label reliability, selecting confident samples, or introducing external semantic priors (Li et al., 2024; Jia et al., 2025; Li et al., 2026b; Jiao et al., 2026). However, these methods still depend heavily on the current representation when propagating sparse supervision. When labeled supports are highly localized and the feature space contains ambiguous or low-density regions, local similarity and model confidence may not faithfully reflect the underlying semantic structure (Iscen et al., 2019; Rizve et al., 2021). Diffusion-based clustering methods can model complex feature distributions (Uziel et al., 2025; Meier et al., 2026), but they mainly use diffusion for unsupervised representation refinement rather than for propagating few-shot semantics. Meanwhile, nonparametric clustering methods can adaptively infer the number of clusters (Ronen et al., 2022; Luo et al., 2025), but they typically estimate latent structure from unlabeled features without explicitly incorporating sparse semantic supervision.

These limitations reveal a central challenge: how to use very limited and partially covered labels to reliably propagate semantic information over a complex unlabeled distribution, while simultaneously recovering the latent cluster structure when its cardinality may be unknown. Sparse labels provide only local semantic anchors, whereas the unlabeled data distribution contains the broader

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.

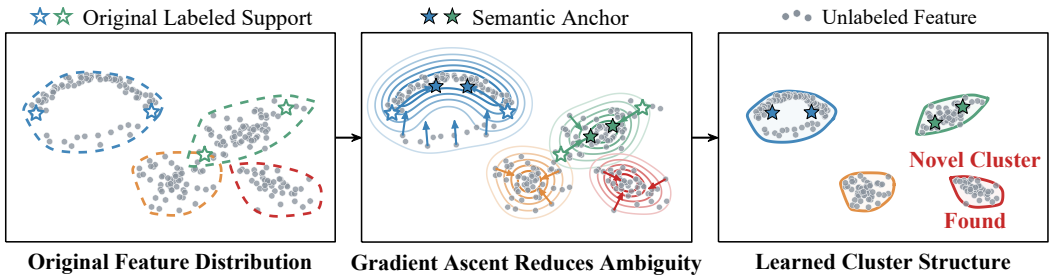


Figure 1: Motivation and intuition of DD-FSC. Sparse labeled supports cannot reliably reach boundary samples or cover all latent categories (left). In response, density-gradient guidance promotes cluster compactness (middle). When the cluster count K is unknown, DP inference adaptively estimates K and sample assignments (right).

geometric structure needed to extend them beyond observed categories. At the same time, cluster-number estimation should remain consistent with the evolving representation rather than being treated as an isolated post-processing step. As illustrated in Figure 1, solving this problem therefore requires both distribution-aware semantic propagation and adaptive cluster-structure inference.

To address this challenge, we propose **DD-FSC**, a nonparametric few-shot clustering framework that jointly models feature-space density and latent cluster structure. DD-FSC uses a diffusion model to estimate the density geometry of the feature distribution and employs score-guided Langevin dynamics to expand sparse semantic anchors toward representative high-density regions, from which reliable same-category samples can be identified. In parallel, Dirichlet process inference adaptively models the latent cluster structure and estimates the effective number of clusters when K is unknown. We further show that the deterministic drift of score-guided Langevin dynamics is positively aligned with the modeled log-density gradient, providing a principled basis for density-guided semantic propagation. In this way, DD-FSC supports few-shot clustering under both known- and unknown-K settings.

Our main contributions are summarized as follows:

- We propose DD-FSC, a unified few-shot clustering framework that jointly models sparse semantic supervision, feature-density structure, and latent cluster structure, supporting both known- and unknown-K settings under partial category coverage.
- We develop diffusion-guided semantic propagation and density-aware representation learning. Score-guided Langevin dynamics explores high-density regions to expand few-shot supervision and construct density-guided positives. We further prove that the learned deterministic drift strictly increases the Gaussian-smoothed feature log-density under explicit conditions on score-estimation error and step size.
- We couple diffusion-based density modeling with a Dirichlet process for adaptive cluster-structure inference without a predefined K. Posterior cluster responsibilities and prototype-level contrastive learning further connect the inferred nonparametric structure with representation learning. Extensive experiments validate the effectiveness of DD-FSC under known-K, unknown-K, and generalized category discovery settings.

2 Related Work

Deep Clustering. Without semantic labels, deep clustering constructs supervisory signals from the data itself. Early approaches exploit self-supervised neighborhood consistency or jointly optimize instance- and cluster-level contrastive objectives (Van Gansbeke et al., 2020; Li et al., 2021). Subsequent methods improve internally generated supervision through confidence calibration, reclustering, sample selection, and local cluster structure (Jia et al., 2025; Miklautz et al., 2025; Li et al., 2026b; Jiao et al., 2026). A complementary line incorporates semantic priors from vision-language models, external knowledge, or small labeled support sets (Cai et al., 2023; Li et al., 2024; Jiu et al., 2026; Zhang et al., 2026b; Li et al., 2025). DD-FSC follows the latter setting but uses supervision differently. Sparse support labels identify category-consistent candidates along score-guided density trajectories, while the resulting representation updates are coupled with adaptive DP inference.

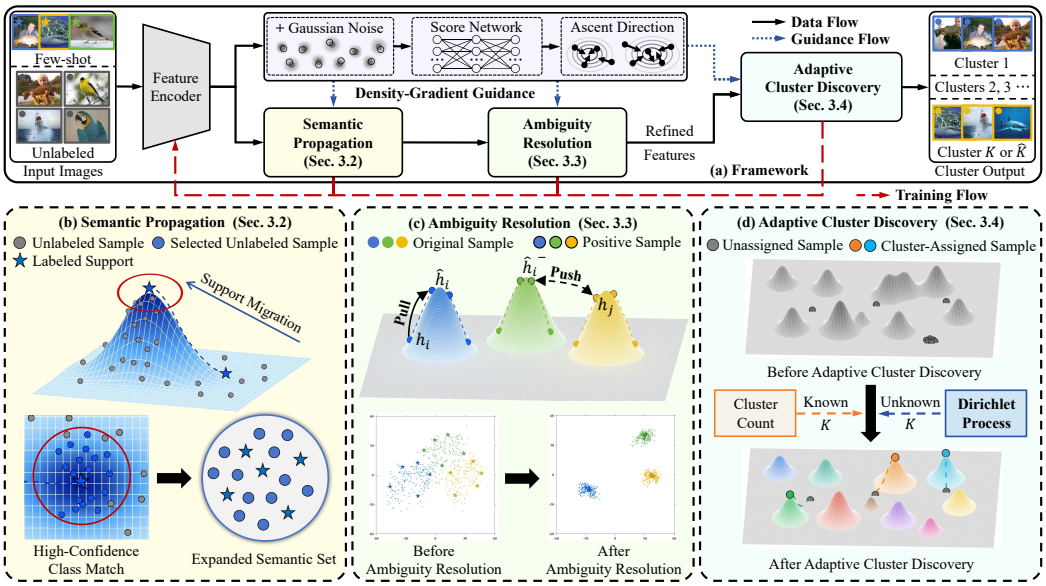


Figure 2: Overview of DD-FSC. (a) Few-shot labeled and unlabeled images are first encoded into features. A noise-conditional score network provides shared density-gradient guidance. Semantic propagation and ambiguity resolution then refine the features. Adaptive cluster discovery finally produces cluster assignments. (b) Semantic propagation follows the density gradients to select high-confidence candidates and expand semantic sets. (c) Ambiguity resolution aligns features with their density-guided counterparts to refine ambiguous samples. (d) Adaptive cluster discovery uses DP responsibilities to guide prototype diffusion and infer K when unknown.

Diffusion for Clustering. Diffusion models provide flexible priors for modeling complex data distributions (Song & Ermon, 2019; Ho et al., 2020; Song et al., 2021). In clustering, these priors have been used to learn generative representations, produce self-supervised assignments, and denoise latent embeddings (Palumbo et al., 2024; Uziel et al., 2025; Meier et al., 2026). These approaches exploit diffusion to recover structure from unlabeled data. Such structure alone does not provide category semantics. DD-FSC combines score-guided density trajectories with few-shot semantic anchors. This enables semantic propagation and representation refinement.

Clustering with Unknown-K. When K is unknown, erroneous assignments can mislead cluster-number estimation, while an inaccurate K further distorts the resulting partition (Ronen et al., 2022; Leiber et al., 2024). Existing methods address this coupling by adapting cluster structure from learned representations. Component-based methods use split–merge updates or remove dying clusters from an overcomplete initialization (Ronen et al., 2022; Leiber et al., 2024). For hyperspherical embeddings, Bayesian nonparametric mixtures infer K while respecting angular geometry (Luo et al., 2025; Tepakbong et al., 2026). Hierarchy-based methods instead select K by maximizing self-supervised consensus (Zhang et al., 2026a). These strategies improve structural adaptation, but ambiguous partitions must still be resolved from unlabeled representations alone. DD-FSC closes this gap by coupling sparse semantic guidance with joint DP–diffusion learning.

3 Method

3.1 Problem Formulation and Overview

We consider few-shot clustering with partial category coverage. Let $\mathcal{D} = \mathcal{D}_l \cup \mathcal{D}_u$ denote a dataset drawn from a latent category space $\mathcal{Y}$, where $K = |\mathcal{Y}|$. The labeled support set is $\mathcal{D}_l = \{(x_i, y_i)\}_{i=1}^{N_l}$, and the unlabeled set is $\mathcal{D}_u = \{x_j\}_{j=1}^{N_u}$. Let $N = N_l + N_u$ denote the total number of samples, and index the pooled inputs as $\{x_i\}_{i=1}^{N}$ when labels are not required. The observed categories satisfy $\mathcal{Y}_l = \{y_i : (x_i, y_i) \in \mathcal{D}_l\} \subsetneq \mathcal{Y}$. Each category in $\mathcal{Y}_l$ provides only a small number of labeled examples, while $\mathcal{D}_u$ contains the remaining samples from these categories and all samples from $\mathcal{Y} \setminus \mathcal{Y}_l$. The goal is to partition $\mathcal{D}$ into K semantic clusters using $\mathcal{D}_l$ as sparse

guidance. In the known-K setting, K is given; in the unknown-K setting, it is inferred jointly with the cluster assignments.

Given this sparse support supervision, DD-FSC couples diffusion-based density modeling with cluster inference, as summarized in Figure 2. Figure 2(a) provides an overview of DD-FSC's inputs, outputs, and end-to-end data flow, while Figure 2(b)–(d) details its three core modules.

3.2 Few-Shot Gradient Ascent for Semantic Propagation

Sparse labeled supports identify the observed categories but provide limited coverage of their feature distributions. To address this limitation, we use a noise-conditional score model to explore high-density regions and construct density-informed anchors for selecting unlabeled candidates. Let $h = f_\phi(x)$ denote an adapted feature. For its noise-σ perturbation $\tilde{h}$, the score network estimates the smoothed marginal score, $s_\psi(\tilde{h}, \sigma) \approx \nabla_{\tilde{h}} \log p_\sigma(\tilde{h})$. This marginal score provides a density direction but no category identity. Few-shot anchors and confidence filtering therefore impose the semantic constraint. At each noise level σ, let $\mathbf{I}$ be the identity matrix, $\gamma > 0$ the exploration scale, and $\xi_i^{(m)} \sim \mathcal{N}(0, \mathbf{I})$. A labeled-feature trajectory takes M steps of size $\epsilon > 0$ as

$$\tilde{h}_i^{(m)} = \tilde{h}_i^{(m-1)} + \epsilon\, s_\psi\Big(\tilde{h}_i^{(m-1)}, \sigma\Big) + \gamma\sqrt{2\epsilon}\,\xi_i^{(m)}, \qquad m = 1, \dots, M. \tag{1}$$

The update combines score guidance with Gaussian exploration. Theorem 1 establishes when its learned drift remains density-increasing despite finite score-estimation error.

Theorem 1. *Fix an encoder state and a noise level $\sigma > 0$. Let μ_{data} denote the distribution of the normalized features and let $p_\sigma(\mathbf{h}) = \int \mathcal{N}(\mathbf{h}; \mathbf{u}, \sigma^2\mathbf{I})\, \mathrm{d}\mu_{\mathrm{data}}(\mathbf{u})$ be its Gaussian-smoothed density. Define $\ell_\sigma(\mathbf{h}) = \log p_\sigma(\mathbf{h})$ and $g_\sigma(\mathbf{h}) = \nabla \ell_\sigma(\mathbf{h})$. Assume that g_σ is L_σ-Lipschitz on a convex neighborhood $\mathcal{U}$ containing the update segment and that the learned score satisfies $\|s_\psi(\mathbf{h}, \sigma) - g_\sigma(\mathbf{h})\|_2 \le \delta_\sigma$ on $\mathcal{U}$. For the update in equation 1, the conditional mean displacement differs from the ideal score drift by at most $\epsilon\delta_\sigma$. Moreover, if $\delta_\sigma < \|g_\sigma(\mathbf{h})\|_2$, the deterministic endpoint $\mathbf{h}_{\mathrm{d}}^+ = \mathbf{h} + \epsilon s_\psi(\mathbf{h}, \sigma)$ satisfies $\ell_\sigma(\mathbf{h}_{\mathrm{d}}^+) - \ell_\sigma(\mathbf{h}) \ge \epsilon\|g_\sigma(\mathbf{h})\|_2(\|g_\sigma(\mathbf{h})\|_2 - \delta_\sigma) - \frac{L_\sigma \epsilon^2}{2}\|s_\psi(\mathbf{h}, \sigma)\|_2^2$. Consequently, the learned drift strictly increases the smoothed log-density whenever $0 < \epsilon < 2\|g_\sigma(\mathbf{h})\|_2(\|g_\sigma(\mathbf{h})\|_2 - \delta_\sigma)/(L_\sigma\|s_\psi(\mathbf{h}, \sigma)\|_2^2)$.*

Theorem 1 provides a finite-step guarantee without assuming an ideal score estimator. It quantifies how score-estimation error and the step size jointly determine whether the learned drift increases the Gaussian-smoothed feature density. The result remains applicable when the normalized clean-feature distribution is supported on a lower-dimensional manifold because p_σ is smooth for every $\sigma > 0$. The Gaussian term preserves exploration, so individual stochastic trajectories need not increase density monotonically. Full assumptions and the proof are provided in Appendix B.1.2.

The terminal features are then used to construct category anchors. Let $\mathcal{D}_l^y = \{x_i : (x_i, y_i) \in \mathcal{D}_l,\ y_i = y\}$ denote the labeled samples from category y. Let $\hat{h}_i$ denote the terminal feature obtained from x_i. For each $y \in \mathcal{Y}_l$, a_y is the normalized mean of its support endpoints. Define $\bar{h}_i = h_i/\|h_i\|_2$ for every sample. For an unlabeled sample x_j, its assignment score with temperature $\tau_f > 0$ is

$$r_{jy} = \frac{\exp\big(\bar{h}_j^\top a_y/\tau_f\big)}{\sum_{c\in\mathcal{Y}_l}\exp\big(\bar{h}_j^\top a_c/\tau_f\big)}. \tag{2}$$

Based on these scores, the predicted category and confidence are $\bar{y}_j = \arg\max_{y\in\mathcal{Y}_l} r_{jy}$ and $c_j = \max_{y\in\mathcal{Y}_l} r_{jy}$, respectively. For each $y \in \mathcal{Y}_l$, $\mathcal{C}_y$ contains up to P highest-confidence candidates satisfying $\bar{y}_j = y$ and $c_j \ge \delta$. The expanded set is $\widehat{\mathcal{D}}_y = \mathcal{D}_l^y \cup \mathcal{C}_y$. This selection couples density structure with sparse semantics. All remaining samples stay unlabeled. Using these expanded sets, the Few-Shot Contrastive Loss (FCL) with temperature $\tau > 0$ is defined as

$$\mathcal{L}_{\mathrm{FCL}} = -\sum_{y\in\mathcal{Y}_l}\sum_{x_i\in\widehat{\mathcal{D}}_y}\log\frac{\sum\limits_{x_j\in\widehat{\mathcal{D}}_y\setminus\{x_i\}}\exp\big(\bar{h}_i^\top\bar{h}_j/\tau\big)}{\sum\limits_{x_k\in\mathcal{D}\setminus\{x_i\}}\exp\big(\bar{h}_i^\top\bar{h}_k/\tau\big)}. \tag{3}$$

FCL treats samples within each expanded set as positives, extending sparse supervision to high-confidence unlabeled samples. Further details on the annealed schedule, anchor construction, candidate selection, and mini-batch implementation are provided in Appendix B.1.3.

3.3 AMBIGUITY RESOLUTION WITH DENSITY-GUIDED POSITIVE-PAIR CONTRAST

The choice of positive pairs strongly shapes the invariances learned by contrastive objectives. Common augmentation-based methods pair views of the same image, primarily encoding transformation invariance. However, such pairs ignore feature-space density, limiting how ambiguous or low-density samples are guided toward high-density regions.

To address this limitation, DD-FSC constructs positive pairs through score-guided feature-space trajectories. Semantic expansion in Section 3.2 uses trajectories initialized from labeled supports. Here, the update in equation 1 is instead applied to every feature $h_i = f_\phi(x_i)$, yielding the endpoint $\hat{h}_i$. The deterministic drift of this trajectory follows the noisy log-density gradient (Theorem 1). Its endpoint thus provides a density-informed counterpart of the original feature. Aligning the pair converts diffusion-derived density information into a direct representation-learning signal. To optimize this alignment over a mini-batch $\mathcal{B}$, with $\mathrm{sg}(\cdot)$ denoting stop-gradient, we define the Density-Guided Contrastive Loss (DGCL) as

$$\mathcal{L}_{\mathrm{DGCL}} = \frac{1}{|\mathcal{B}|} \sum_{x_i \in \mathcal{B}} \left\| h_i - \mathrm{sg}(\hat{h}_i) \right\|_2^2 . \tag{4}$$

This makes $\hat{h}_i$ a fixed target for the encoder. DGCL aligns each feature with its density-guided counterpart. It neither determines class identity nor contains explicit negatives. Semantic and prototype-level separation are instead encouraged by $\mathcal{L}_{\mathrm{FCL}}$ and $\mathcal{L}_{\mathrm{PCL}}$, respectively. Together, these objectives combine sparse semantic guidance with density-aware representation refinement. Further details on trajectory construction and gradient flow are provided in Appendix B.1.4.

3.4 ADAPTIVE CLUSTER DISCOVERY VIA DP–DIFFUSION COUPLING

The density-guided alignment above refines features but does not assign all samples to clusters. When K is unknown, assignments and the cluster count must adapt as features evolve. Without coupling, inferred clusters cannot guide representation updates, and updated features cannot refine cluster inference. DD-FSC therefore couples diffusion-based density guidance with Dirichlet process (DP) inference. DP responsibilities guide diffusion trajectories, and updated representations refine the DP posterior. This feedback jointly refines representations and the inferred partition.

Let $H = \{h_i\}_{i=1}^N$ be the feature set, Q the variational posterior, π_k component k's mixing weight, and $\Theta = \{(\mu_k, \Sigma_k)\}_{k=1}^{K_{\max}}$ the component means and diagonal covariances. DD-FSC fits a truncated stick-breaking DP mixture with at most $K_{\max}$ components by maximizing the evidence lower bound (ELBO), $\mathcal{L}_{\mathrm{ELBO}}$. For known-$K$, $K_{\max} = K$ and all components remain active. For unknown-K, $K_{\max}$ is an upper bound. Given activation threshold $\varepsilon > 0$, components satisfying $\mathbb{E}_Q[\pi_k] \geq \varepsilon$ form the active set $\mathcal{A}$, giving $\widehat{K} = |\mathcal{A}|$. Thus, $\widehat{K}$ adapts as the posterior masses change.

To inject this partition into diffusion, let $t_i^{(m)}$ be sample i's state after m updates and $\rho_{ik}^{(m)}$ the local responsibility of active component k at state $t_i^{(m)}$. The posterior-weighted guidance field is

$$F_{\mathrm{DP}}(t_i^{(m)}; \Theta) = \sum_{k \in \mathcal{A}} \rho_{ik}^{(m)} \mathbb{E}_Q \left[\Sigma_k^{-1} (\mu_k - t_i^{(m)}) \right] . \tag{5}$$

This field averages precision-weighted directions to component centers using local responsibilities. Because its magnitude depends on the posterior precision scale, let $\varepsilon_{\mathrm{norm}} > 0$ be a numerical stability constant and use the bounded direction $\overline{F}_{\mathrm{DP}} = F_{\mathrm{DP}} / \max\{\|F_{\mathrm{DP}}\|_2, \varepsilon_{\mathrm{norm}}\}$. For step size $\eta_m > 0$, noise $\xi_i^{(m)} \sim \mathcal{N}(0, \mathbf{I})$, and DP-guidance weight $\lambda_{\mathrm{DP}} \geq 0$, combining this direction with the marginal score gives

$$t_i^{(m+1)} = t_i^{(m)} + \eta_m \left[s_\psi(t_i^{(m)}, \sigma_m) + \lambda_{\mathrm{DP}} \overline{F}_{\mathrm{DP}}(t_i^{(m)}; \Theta) \right] + \gamma \sqrt{2\eta_m}\, \xi_i^{(m)}, \tag{6}$$

Since $\|\overline{F}_{\mathrm{DP}}\|_2 \leq 1$, the DP contribution to each deterministic update is bounded by $\eta_m \lambda_{\mathrm{DP}}$. The score term follows the estimated marginal density gradient, whereas the normalized DP field injects the inferred partition without inheriting its absolute precision scale. The resulting trajectory is therefore both density- and cluster-aware.

The inferred structure also guides representation learning through prototypes. For a mini-batch $\mathcal{B}$, let $\mathcal{A}_{\mathcal{B}}$ contain components with sufficient posterior mass and $\mathbf{c}_k$ denote their normalized posterior-weighted prototypes. Applying equation 6 to $\mathbf{c}_k$ and normalizing the endpoint yields $\mathbf{c}'_k$. With temperature $\tau_p > 0$, the Prototype Contrastive Loss (PCL) is

$$\mathcal{L}_{\mathrm{PCL}} = -\frac{1}{|\mathcal{A}_{\mathcal{B}}|} \sum_{k \in \mathcal{A}_{\mathcal{B}}} \log \frac{\exp\big(\mathbf{c}_k^\top \operatorname{sg}(\mathbf{c}'_k)/\tau_p\big)}{\exp\big(\mathbf{c}_k^\top \operatorname{sg}(\mathbf{c}'_k)/\tau_p\big) + \sum_{\substack{j \in \mathcal{A}_{\mathcal{B}} \\ j \neq k}} \exp\big(\mathbf{c}_k^\top \mathbf{c}_j/\tau_p\big)}. \tag{7}$$

PCL treats $\mathbf{c}'_k$ as the positive for $\mathbf{c}_k$ and the other active prototypes as negatives. It thereby promotes cluster-aware alignment and inter-cluster separation.

Let $\mathcal{L}_{\mathrm{score}}$ be the score-matching loss, d_h the feature dimension, and $\overline{\mathcal{L}}_{\mathrm{ELBO}} = \mathcal{L}_{\mathrm{ELBO}}/(Nd_h)$ the normalized ELBO. For nonnegative weights $\lambda_{\mathrm{score}}, \lambda_{\mathrm{ELBO}}, \lambda_{\mathrm{FCL}}, \lambda_{\mathrm{DGCL}}, \lambda_{\mathrm{PCL}}$, we minimize

$$\mathcal{L}_{\mathrm{total}} = \lambda_{\mathrm{score}}\mathcal{L}_{\mathrm{score}} - \lambda_{\mathrm{ELBO}}\overline{\mathcal{L}}_{\mathrm{ELBO}} + \lambda_{\mathrm{FCL}}\mathcal{L}_{\mathrm{FCL}} + \lambda_{\mathrm{DGCL}}\mathcal{L}_{\mathrm{DGCL}} + \lambda_{\mathrm{PCL}}\mathcal{L}_{\mathrm{PCL}}, \tag{8}$$

The negative sign reflects ELBO maximization. Variational and neural updates alternate, with responsibilities fixed during each neural update. Appendix B.2 details DP inference, Appendix C gives the complete alternating optimization procedure, and Appendix I visualizes learned representations and diffusion-guided retrieval.

4 Experiments

4.1 Experimental Setup

Datasets and Evaluation Metrics. The experimental analyses are conducted on ten image clustering benchmarks. The main evaluation includes four ImageNet benchmarks, while six additional benchmarks are considered in Appendix D. Detailed descriptions and statistics for these benchmarks are provided in Appendix A. We report normalized mutual information (NMI), clustering accuracy (ACC), and adjusted Rand index (ARI) as percentages, with higher values indicating better performance. For cluster-count estimation, we report $\widehat{K}$ and measure the absolute deviation of the mean estimate from the ground-truth K, with a smaller deviation indicating a more accurate estimate.

Protocol. We consider both known- and unknown-K settings. In the known-K setting, the ground-truth cluster count is provided to the clustering procedure. In the unknown-K setting, it is withheld and inferred using the Dirichlet process. Label-guided DD-FSC uses a partial-category five-shot protocol: approximately 30% of the categories are labeled, with five labeled samples per selected category. Categories and supports are independently resampled over three runs, and means are reported. Appendix C.1 gives the complete procedure, Appendix F reports independent ten-run statistics, and Appendix G evaluates robustness to backbone and supervision choices.

Baselines. For **known-K**, we compare DD-FSC-U (the fully unsupervised variant of DD-FSC) with MiCE (Tsai et al., 2021), TAC (Li et al., 2024), CPP (Chu et al., 2024), CLUDI (Uziel et al., 2025), MiniClustering (Li et al., 2026b), CDC+SANE (Jiao et al., 2026), and GCPM-G (Zhong et al., 2026). We also compare DD-FSC with Soft k-Means (Ren et al., 2018), cluster-FSL (Ling et al., 2022), UNEM (Zhou et al., 2025), EfficientFSL (Liao et al., 2026), TIM++ (Li et al., 2026a), and OSESSL (Cui et al., 2026). For **unknown-K**, we compare DD-FSC with DeepDPM (Ronen et al., 2022), UNSEEN+DCN (Leiber et al., 2024), SpherePair (Zhang & Chen, 2025), SCMax (Zhang et al., 2026a), and DeepDP-TGMM (Tepakbong et al., 2026). We use published baseline results when available. Otherwise, we reproduce them using public implementations and reported settings.

4.2 Comparison with State-of-the-Art Methods

Known-K clustering. Table 1 compares DD-FSC under the unsupervised and partial-category five-shot protocols. Without labeled supports, DD-FSC-U achieves the best ACC and ARI on three of the four benchmarks, showing that density-guided representation learning remains effective without semantic supervision. With only five labeled samples from 30% of the categories, DD-FSC achieves the best ACC on all four benchmarks, outperforming the strongest baseline by 1.4–4.5 points. It also ranks first or second in NMI throughout. These results indicate that density-aware semantic propagation and positive-pair refinement effectively exploit sparse labels while preserving clustering over the complete category space.

Table 1: Known-K clustering results on ImageNet benchmarks. DD-FSC-U denotes the fully unsupervised variant of DD-FSC. Best results are in **bold** and second best are underlined.

Methods	ImageNet-50			ImageNet-100			ImageNet-200			ImageNet-1K		
	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI
(a) Unsupervised clustering												
MiCE (ICLR, 2021)	85.7	77.2	74.3	82.7	68.3	58.7	80.2	66.1	53.4	68.8	38.5	26.3
TAC (ICML, 2024)	83.7	78.7	77.6	82.1	73.8	67.1	80.9	69.2	60.2	79.4	58.1	43.5
CPP (ICLR, 2024)	89.3	80.6	79.2	88.5	78.2	68.2	85.1	68.4	64.8	**83.8**	67.2	45.9
CLUDI (ICML, 2025)	91.2	82.1	76.2	87.1	76.6	67.8	86.1	73.7	63.2	79.2	68.3	44.8
MiniClustering (ICLR, 2026)	91.8	83.4	78.8	89.3	79.5	65.9	85.7	74.2	64.8	77.1	68.6	46.7
CDC+SANE (ICLR, 2026)	92.9	82.2	81.2	87.8	78.5	69.1	86.2	76.3	63.5	83.3	67.6	47.8
GCPM-G (CVPR, 2026)	**93.3**	85.9	82.2	88.8	**82.3**	71.2	86.4	80.4	**68.3**	79.3	69.3	47.7
DD-FSC-U (Ours)	90.8	**88.2**	**85.6**	**89.5**	82.2	**75.9**	**87.8**	**81.9**	67.7	82.4	**70.3**	**48.3**
(b) Few-shot clustering (30% of categories, 5 shots each)												
Soft k-Means (ICLR, 2018)	86.3	80.8	72.1	84.3	74.7	64.2	78.6	69.1	50.3	65.1	47.2	31.4
cluster-FSL (CVPR, 2022)	87.5	83.1	71.6	85.7	77.4	71.3	83.1	69.7	61.5	73.1	52.7	33.9
UNEM (CVPR, 2025)	91.2	84.3	82.9	88.2	80.1	70.9	85.2	75.4	67.8	75.4	58.9	38.4
EfficientFSL (AAAI, 2026)	93.6	83.6	78.6	89.7	80.7	69.5	85.8	76.6	66.1	78.9	65.2	44.1
TIM++ (AAAI, 2026)	94.9	85.1	81.6	**92.9**	83.6	70.6	87.5	78.2	69.6	81.8	69.2	**52.4**
OSESSL (ICLR, 2026)	94.8	89.9	**89.1**	91.2	83.8	72.4	**89.2**	79.2	70.7	84.7	70.1	51.3
DD-FSC (Ours)	**95.5**	**93.4**	87.3	92.4	**85.2**	**77.4**	88.7	**83.7**	**71.4**	**86.4**	**72.4**	51.2

Table 2: Unknown-K clustering results on ImageNet benchmarks. Best results are in **bold** and second best are underlined.

Methods	ImageNet-50			ImageNet-100			ImageNet-200			ImageNet-1K		
	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI
DeepDPM (CVPR, 2022)	74.2	65.8	51.2	61.6	39.0	24.6	43.0	37.2	15.8	46.6	22.9	18.6
UNSEEN+DCN (ICDMW, 2024)	79.6	68.4	54.8	60.6	41.9	35.7	46.1	42.5	26.6	43.2	34.5	21.6
SpherePair (NeurIPS, 2025)	85.3	66.7	57.1	66.8	46.9	39.8	56.9	41.0	24.5	57.6	33.5	20.5
SCMax (AAAI, 2026)	82.2	70.4	62.0	67.6	47.5	40.3	53.4	38.7	30.3	59.1	32.8	26.7
DeepDP-TGMM (CVPR, 2026)	**86.2**	69.0	65.6	68.9	52.6	**46.2**	59.8	45.4	32.7	57.1	35.3	**29.0**
DD-FSC (Ours)	82.9	**76.0**	**66.8**	**69.7**	**56.2**	44.7	**63.6**	**46.7**	**35.3**	**60.5**	**36.6**	28.1

Table 3: Estimated cluster count $\widehat{K}$ on ImageNet benchmarks under the unknown-K setting. Best results are in **bold** and second best are underlined.

Methods	ImageNet-50	ImageNet-100	ImageNet-200	ImageNet-1K
DeepDPM (CVPR, 2022)	$\mathbf{51.3 \pm 1.2}$	89.3 ± 3.1	158.0 ± 3.7	1218.7 ± 18.8
UNSEEN+DCN (ICDMW, 2024)	40.3 ± 2.5	107.7 ± 3.7	185.0 ± 2.8	901.3 ± 15.4
SpherePair (NeurIPS, 2025)	53.7 ± 1.7	95.0 ± 15.0	196.7 ± 13.6	865.0 ± 19.5
SCMax (AAAI, 2026)	43.7 ± 3.1	115.3 ± 6.0	211.7 ± 4.0	1098.7 ± 26.1
DeepDP-TGMM (CVPR, 2026)	47.3 ± 1.9	105.3 ± 2.1	$\mathbf{201.0 \pm 5.7}$	1063.7 ± 12.1
DD-FSC (Ours)	51.7 ± 0.9	$\mathbf{103.0 \pm 1.4}$	198.7 ± 1.9	$\mathbf{1049.0 \pm 10.2}$

Unknown-K clustering and cluster-count estimation. Tables 2 and 3 evaluate the two defining aspects of latent-structure recovery: sample membership and category cardinality. DD-FSC achieves the best ACC on all four benchmarks and the best NMI on three, while its $\widehat{K}$ is closest or second-closest to the ground truth, with the lowest variance throughout. Together, these results show that DD-FSC jointly recovers cluster memberships and the cardinality of the latent partition. This joint recovery reflects its bidirectional design, in which density-guided representation learning and nonparametric partition inference adapt to each other rather than being optimized in separate stages.

4.3 Ablation Study

We conduct controlled ablations on ImageNet-100 under the partial-category five-shot protocol. As summarized in Table 4, the known-K setting isolates the effects of semantic expansion and the training objectives, whereas the unknown-K setting further examines the representation and nonparametric inference components. Reporting $\widehat{K}$ in the latter setting distinguishes improvements in assignment quality from those in latent-structure recovery. Additional implementation details and benchmark-level ablations are provided in Appendix E.

Table 4: Ablation study of DD-FSC on ImageNet-100.

(a) Known-K setting

Variant	NMI	ACC	ARI
Semantic-expansion controls			
Support only	78.2	64.7	52.2
kNN expansion	82.4	66.4	54.9
Noise only	81.7	66.3	55.4
Score only	84.4	71.7	61.7
Score + filtering	88.2	77.3	68.4
Training-objective ablations			
w/o FCL	89.5	82.2	75.9
w/o DGCL	87.1	76.9	67.7
w/o PCL	89.5	80.3	71.0
Full DD-FSC	92.4	85.2	77.4

(b) Unknown-K setting

Variant	NMI	ACC	ARI	$\widehat{K}$
Representation-learning ablations				
w/o score guidance	62.4	44.1	31.7	79.3±5.9
w/o semantic expansion	63.5	46.3	34.4	120.0±2.9
w/o FCL	63.2	46.9	35.8	118.7±1.7
w/o DGCL	65.4	49.8	38.1	109.0±3.6
Adaptive-clustering ablations				
w/o joint DP learning	62.3	42.2	30.2	86.7±5.8
w/o DP–diffusion coupling	65.1	47.8	36.6	112.7±1.2
w/o adaptive updates	64.5	48.2	36.8	123.0±2.9
w/o PCL	66.1	51.7	40.4	107.7±3.9
Full DD-FSC	69.7	56.2	44.7	103.0±1.4

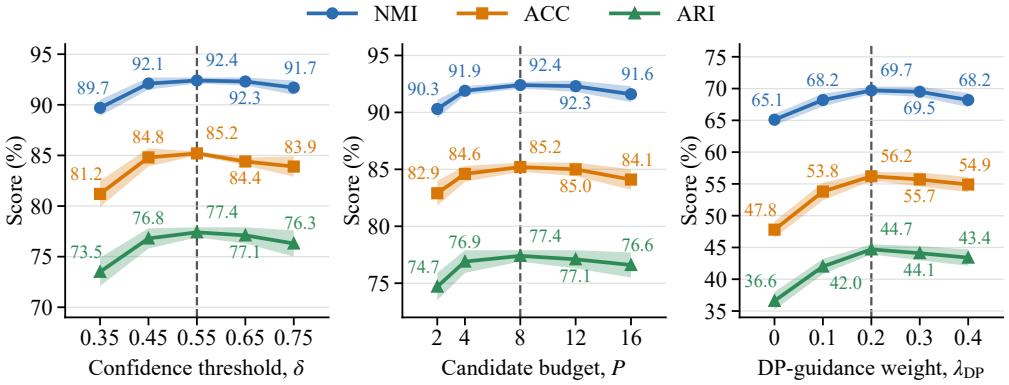


Figure 3: Sensitivity of DD-FSC to its core hyperparameters on ImageNet-100. The left and middle panels evaluate δ and P under known-K, while the right panel evaluates λ_{DP} under unknown-K. Curves and shaded regions denote the mean and one standard deviation over three prespecified seeds, and dashed lines mark the default values.

Table 4 reports that under **known-K**, score-guided expansion outperforms kNN expansion and noise-only sampling, while confidence filtering under the same top-P cap rejects low-confidence candidates and further improves all three metrics. This ordering attributes the gain to learned density ascent and reliable semantic selection. Removing DGCL causes the largest objective-level drop, showing that density-guided positive pairs are crucial for refining the representation geometry. For **unknown-K**, removing score guidance or joint DP learning causes the largest degradation and underestimates K, consistent with cluster merging. In contrast, all other ablations overestimate it, consistent with cluster fragmentation. Larger cluster-count errors without DP–diffusion coupling or adaptive updates indicate that density evolution and nonparametric partitioning require coordination. Accordingly, the full model achieves the best clustering results and the most accurate estimate of K.

4.4 Hyperparameter Sensitivity

We evaluate the sensitivity of DD-FSC to three core hyperparameters that directly control semantic expansion and DP–diffusion coupling. The confidence threshold δ determines the minimum category-assignment confidence required for admitting an unlabeled candidate, while the candidate budget P limits the number of accepted candidates per observed category. We evaluate both parameters under known-K to isolate supervision utilization from cluster-count estimation. Under unknown-K, the DP-guidance weight λ_{DP} controls the contribution of the inferred cluster structure to the diffusion trajectory. Each parameter is varied independently on ImageNet-100 while all remaining settings are fixed. Results are reported in Figure 3 as the mean and one standard deviation over three prespecified seeds. Additional analyses of annotation conditions, score-guided exploration, objective weights, posterior activation, and truncation capacity are provided in Appendix H.

Figure 3 shows that all three default values achieve the highest mean and the lowest standard deviation for NMI, ACC, and ARI. Varying δ changes the three metrics by at most 2.7, 4.0, and 3.9 points,

Table 5: Additional evaluation of DD-FSC-GCD on generalized category discovery (GCD). DD-FSC-GCD denotes DD-FSC adapted to the standard GCD protocol. Accuracy (%) is reported for All, Old, and New categories. Best results are in **bold** and second best are underlined.

(a) Clustering with the ground-truth number of categories K given

Methods	CIFAR-100			ImageNet-100			CUB-200			Stanford Cars			Herbarium19		
	All	Old	New	All	Old	New	All	Old	New	All	Old	New	All	Old	New
GCD (CVPR, 2022)	73.0	76.2	66.5	74.1	89.8	66.3	51.3	56.6	48.7	39.0	57.6	29.9	35.4	51.0	27.0
CMS (CVPR, 2024)	82.3	85.7	75.5	84.7	95.6	79.2	68.2	76.5	64.0	56.9	76.1	47.6	36.4	54.9	26.4
NC-GCD (NeurIPS, 2025)	82.7	85.5	77.3	88.4	94.1	85.5	74.8	76.8	73.8	59.9	77.8	51.2	46.4	58.4	40.7
FAIR (AAAI, 2026)	84.2	83.0	**86.6**	86.0	94.4	81.7	73.7	75.3	73.0	63.6	70.4	60.4	49.1	59.1	43.8
CMS+AL-GCD (CVPR, 2026)	83.9	84.9	82.1	87.6	95.9	84.0	73.8	**79.0**	71.2	80.0	**90.3**	75.0	39.6	59.2	29.0
DD-FSC-GCD (Ours)	**88.5**	**92.7**	80.1	**90.8**	**97.5**	**87.4**	**76.6**	71.9	**78.9**	**81.1**	88.2	**77.7**	**54.0**	**61.1**	**50.2**

(b) Clustering without the ground-truth number of categories K given

Methods	CIFAR-100			ImageNet-100			CUB-200			Stanford Cars			Herbarium19		
	All	Old	New	All	Old	New	All	Old	New	All	Old	New	All	Old	New
GCD (CVPR, 2022)	70.8	77.6	57.0	77.9	91.1	71.3	51.1	56.4	48.4	39.1	58.6	29.7	37.2	51.7	29.4
CMS (CVPR, 2024)	79.6	83.2	72.3	81.3	95.6	74.2	64.4	68.2	62.2	51.7	68.9	43.4	37.4	56.5	27.1
NC-GCD (NeurIPS, 2025)	80.5	83.7	74.0	**85.7**	95.9	80.6	70.3	72.1	69.4	54.0	73.1	44.8	42.3	56.2	34.8
FAIR (AAAI, 2026)	83.2	82.6	84.4	84.8	92.9	**80.7**	72.9	71.1	**73.8**	60.9	72.4	55.4	47.6	56.9	42.7
CMS+AL-GCD (CVPR, 2026)	83.3	**84.9**	80.2	**85.7**	96.5	80.3	73.7	79.1	71.0	79.9	**90.5**	74.7	41.2	**59.0**	31.6
DD-FSC-GCD (Ours)	**84.5**	83.8	**86.0**	84.8	**97.5**	78.4	**74.2**	**80.9**	70.9	**80.0**	88.6	**75.8**	**51.3**	57.2	**48.1**

respectively. The decline at either end reflects the reliability–coverage trade-off: a low threshold can admit uncertain candidates, whereas a high threshold restricts semantic expansion. Varying P produces corresponding ranges of 2.1, 2.3, and 2.7 points. Performance rises to $P = 8$ and then decreases gradually, showing that sufficient candidate coverage is beneficial while an excessive budget can introduce lower-ranked candidates. Under unknown-K, setting $\lambda_{\mathrm{DP}} = 0$ removes cluster-aware guidance and reduces NMI, ACC, and ARI by 4.6, 8.4, and 8.1 points relative to the default 0.2. The corresponding sensitivity of the inferred cluster count to λ_{DP} is reported in Appendix H.3 (Figure 9). This clear gain shows that the inferred DP structure provides information complementary to density guidance. The nearby value 0.3 remains within 0.6 points on all three metrics, indicating a stable region around the default. Together, these results validate both controlled semantic expansion and DP–diffusion coupling without requiring exact parameter tuning.

4.5 Additional Experiments on Generalized Category Discovery

To assess DD-FSC's broader applicability, we further evaluate it on five generalized category discovery (GCD) benchmarks. Following the standard GCD protocol, we adapt DD-FSC using labeled data from Old categories and unlabeled data from both Old and New categories, yielding DD-FSC-GCD. Dataset details are provided in Appendix A. We compare DD-FSC-GCD with GCD (Vaze et al., 2022), CMS (Choi et al., 2024), NC-GCD (Han et al., 2025), FAIR (Bian et al., 2026), and CMS+AL-GCD (Han et al., 2026) in Table 5. For **known-K**, DD-FSC-GCD ranks first on 12 of the 15 metrics, outperforming the strongest baseline by 7.0, 5.1, and 6.4 points on CIFAR-100 Old, CUB-200 New, and Herbarium19 New, respectively. For **unknown-K**, it ranks first on 9 metrics and achieves the best All-category accuracy on four of the five benchmarks. Together with our main few-shot results, these findings show that DD-FSC's core design extends to GCD and remains effective under both known- and unknown-K, demonstrating robustness across clustering protocols.

5 Conclusion

This work proposes DD-FSC, a unified few-shot clustering framework for partial category coverage with known and unknown cluster numbers. It employs a noise-conditional score model and Langevin dynamics to capture feature density, expand sparse-label semantics, and build density-guided positive pairs. DP posterior inference provides adaptive cluster supervision and removes the need for predefined cluster counts. Experiments verify its robust clustering performance across settings and reliable estimation of the latent cluster count.

AI USE STATEMENT

In this work, we used generative AI tools to assist with grammar correction and phrasing refinement. These tools also offered guidance on improving the clarity, consistency, and exposition of the methodological and experimental descriptions, as well as on the organization, completeness, and overall readability of the manuscript. We additionally used generative AI tools for limited code debugging, including assistance with interpreting error messages and identifying potential implementation issues. Generative AI tools were not used to generate research ideas, design experiments, or produce experimental results. All AI-assisted text and code suggestions were reviewed and verified for correctness by the authors. We take full responsibility for the final content of this work.

ETHICS STATEMENT

This paper involves no human subjects, no sensitive data, and no potentially harmful applications. All datasets used are public benchmark datasets. We adhere to ethical guidelines in machine learning research, including proper citation and reproducibility.

REPRODUCIBILITY STATEMENT

The main text presents the learning objectives and explains how DD-FSC is trained and evaluated. Section 4.1 defines the experimental protocol and evaluation metrics, while Appendix A documents the benchmark composition and split statistics. The density-ascent result stated in Theorem 1 is proved in Appendix B.1.2. To make the complete workflow explicit, Appendix C.1 provides the training and inference pseudocode. The architecture, optimization settings, and fixed experimental choices required for implementation are recorded in Appendix C.2. We further characterize the method's practical requirements in Appendix C.3, which reports asymptotic complexity together with measured runtime and memory use. Appendix F evaluates run-to-run variation and statistical significance. The effects of backbone and supervision choices are studied in Appendix G, while Appendix I examines representation evolution and diffusion-guided retrieval. We will make the source code available to facilitate reproduction.

REFERENCES

Yuwei Bian, Shidong Wang, Yazhou Yao, and Haofeng Zhang. Foundation-Adaptive Integrated Refinement for Generalized Category Discovery. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 2453–2461, 2026.

Shaotian Cai, Liping Qiu, Xiaojun Chen, Qin Zhang, and Longteng Chen. Semantic-Enhanced Image Clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 6869–6878, 2023.

Kaidi Cao, Maria Brbic, and Jure Leskovec. Open-World Semi-Supervised Learning. In *International Conference on Learning Representations*, 2022.

Jianlong Chang, Lingfeng Wang, Gaofeng Meng, Shiming Xiang, and Chunhong Pan. Deep Adaptive Image Clustering. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

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, 2024.

Tianzhe Chu, Shengbang Tong, Tianjiao Ding, Xili Dai, Benjamin David Haeffele, Rene Vidal, and Yi Ma. Image Clustering via the Principle of Rate Reduction in the Age of Pretrained Models. In *The Twelfth International Conference on Learning Representations*, 2024.

Adam Coates, Andrew Ng, and Honglak Lee. An Analysis of Single-Layer Networks in Unsupervised Feature Learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 215–223. PMLR, 2011.

Haowen Cui, Shuo Chen, Jun Li, and Jian Yang. One-Shot Exemplars for Class Grounding in Self-Supervised Learning. In *The Fourteenth International Conference on Learning Representations*, 2026.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

Li Deng. The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

Jizhou Han, Shaokun Wang, Yuhang He, Chenhao Ding, Qiang Wang, Xinyuan Gao, SongLin Dong, and Yihong Gong. Consistent Supervised-Unsupervised Alignment for Generalized Category Discovery. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Jizhou Han, Chenhao Ding, Yuhang He, Qiang Wang, Shaokun Wang, SongLin Dong, and Yihong Gong. Learning Like Humans: Analogical Concept Learning for Generalized Category Discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28734–28743, 2026.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

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)*, 2019.

Yuheng Jia, Jianhong Cheng, Hui Liu, and Junhui Hou. Towards Calibrated Deep Clustering Network. In *The Thirteenth International Conference on Learning Representations*, 2025.

Zhengxing Jiao, Yaxin Hou, Jun Ma, Yuhang Li, Ding Ding, Yuheng Jia, Hui Liu, and Junhui Hou. Samples Are Not Equal: A Sample Selection Approach for Deep Clustering. In *The Fourteenth International Conference on Learning Representations*, 2026.

Zizheng Jiu, Feijiang Li, Jieting Wang, Yuhua Qian, and Lu Chen. Spatial Structure and Selective Text Jointly Facilitate Image Clustering. In *The Fourteenth International Conference on Learning Representations*, 2026.

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*, 2013.

Alex Krizhevsky. Learning Multiple Layers of Features from Tiny Images. Technical report, University of Toronto, 2009.

Collin Leiber, Niklas Strauß, Matthias Schubert, and Thomas Seidl. Dying Clusters Is All You Need - Deep Clustering With an Unknown Number of Clusters. In *2024 IEEE International Conference on Data Mining Workshops (ICDMW)*, pp. 726–733, 2024.

Chunming Li, Shidong Wang, and Haofeng Zhang. Few-shot Novel Category Discovery. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pp. 1350–1358. International Joint Conferences on Artificial Intelligence Organization, 2025. Main Track.

Yingping Li, Yutong Zou, Yunshi Huang, Changzhe Jiao, Xinlin Wang, Shen Peng, Zhang Guo, and Shuiping Gou. TIM++: Transductive Information Maximization for Few-Shot CLIP. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 6671–6680, 2026a.

Yunfan Li, Peng Hu, Zitao Liu, Dezhong Peng, Joey Tianyi Zhou, and Xi Peng. Contrastive Clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8547–8555, 2021.

Yunfan Li, Peng Hu, Dezhong Peng, Jiancheng Lv, Jianping Fan, and Xi Peng. Image Clustering with External Guidance. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 27890–27902. PMLR, 2024.

Zhixin Li, Yuheng Jia, Guanliang Chen, Hui Liu, and Junhui Hou. Mini-cluster Guided Long-tailed Deep Clustering. In *The Fourteenth International Conference on Learning Representations*, 2026b.

Wenwen Liao, Hang Ruan, Jianbo Yu, Bing Song, Yuansong Wang, and Xiaofeng Yang. EfficientFSL: Enhancing Few-Shot Classification via Query-Only Tuning In Vision Transformers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 23478–23486, 2026.

Jie Ling, Lei Liao, Meng Yang, and Jia Shuai. Semi-Supervised Few-Shot Learning via Multi-Factor Clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14564–14573, 2022.

Zhiwen Luo, Wentao Fan, Manar Amayri, and Nizar Bouguila. Dynamic Deep Clustering of High-Dimensional Directional Data via Hyperspherical Embeddings with Bayesian Nonparametric Mixtures. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, pp. 938–949, 2025.

Dominik Meier, Shixing Yu, Sagnik Nandy, Promit Ghosal, and Kyra Gan. Clustering by Denoising: Latent plug-and-play diffusion for single-cell embeddings. In *The Fourteenth International Conference on Learning Representations*, 2026.

Lukas Miklautz, Timo Klein, Kevin Sidak, Collin Leiber, Thomas Lang, Andrii Shkabrii, Sebastian Tschiatschek, and Claudia Plant. Breaking the Reclustering Barrier in Centroid-based Deep Clustering. In *The Thirteenth International Conference on Learning Representations*, 2025.

Emanuele Palumbo, Laura Manduchi, Sonia Laguna, Daphné Chopard, and Julia E Vogt. Deep Generative Clustering with Multimodal Diffusion Variational Autoencoders. In *The Twelfth International Conference on Learning Representations*, 2024.

Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B. Tenenbaum, Hugo Larochelle, and Richard S. Zemel. Meta-Learning for Semi-Supervised Few-Shot Classification. In *International Conference on Learning Representations*, 2018.

Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. In Defense of Pseudo-Labeling: An Uncertainty-Aware Pseudo-label Selection Framework for Semi-Supervised Learning. In *International Conference on Learning Representations*, 2021.

Mamshad Nayeem Rizve, Navid Kardan, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. OpenLDN: Learning to Discover Novel Classes for Open-World Semi-Supervised Learning. In *European Conference on Computer Vision*, pp. 382–401, 2022.

Meitar Ronen, Shahaf E. Finder, and Oren Freifeld. DeepDPM: Deep Clustering With an Unknown Number of Clusters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9861–9870, 2022.

Yang Song and Stefano Ermon. Generative Modeling by Estimating Gradients of the Data Distribution. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-Based Generative Modeling through Stochastic Differential Equations. In *International Conference on Learning Representations*, 2021.

Kiat Chuan Tan, Yulong Liu, Barbara Ambrose, Melissa Tulig, and Serge Belongie. The Herbarium Challenge 2019 Dataset. arXiv preprint arXiv:1906.05372, 2019.

Cyril Kana Tepakbong, Kévin Bouchard, and Julien Maitre. DeepDP-TGMM: Amortized Non-Parametric Clustering for Hyperspherical Self-Supervised Representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 7483–7492, 2026.

Tsung Wei Tsai, Chongxuan Li, and Jun Zhu. MiCE: Mixture of Contrastive Experts for Unsupervised Image Clustering. In *International Conference on Learning Representations*, 2021.

Roy Uziel, Irit Chelly, Oren Freifeld, and Ari Pakman. Clustering via Self-Supervised Diffusion. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 60711–60726. PMLR, 2025.

Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. SCAN: Learning to Classify Images Without Labels. In *Computer Vision – ECCV 2020*, pp. 268–285. Springer, 2020.

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.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. arXiv preprint arXiv:1708.07747, 2017.

Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised Deep Embedding for Clustering Analysis. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 478–487. PMLR, 2016.

Lijun Zhang, Suyuan Liu, Siwei Wang, Shengju Yu, Xueling Zhu, Miaomiao Li, and Xinwang Liu. Parameter-Free Clustering via Self-Supervised Consensus Maximization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 28310–28318, 2026a.

Shaojie Zhang and Ke Chen. Angular Constraint Embedding via SpherePair Loss for Constrained Clustering. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Xiaohan Zhang, Chao Zhang, Deng Xu, Hong YU, Chunlin Chen, and Huaxiong Li. Semantic-Augmented Image Clustering via Adaptive Multi-Modal Collaboration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 28437–28445, 2026b.

Wenliang Zhong, Rob Barton, Lucas Goncalves, Kushal Kumar, Feng Jiang, Hehuan Ma, Yuzhi Guo, Vidit Bansal, Karim Bouyarmane, and Junzhou Huang. Universal Guideline-Driven Image Clustering via a Hybrid LLM Agent. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 33749–33760, 2026.

Long Zhou, Fereshteh Shakeri, Aymen Sadraoui, Mounir Kaaniche, Jean-Christophe Pesquet, and Ismail Ben Ayed. UNEM: UNrolled Generalized EM for Transductive Few-Shot Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9665–9675, 2025.

APPENDIX CONTENTS

A Benchmark Datasets **15**

B Additional Methodological and Theoretical Details **16**

B.1 Score-Guided Semantic Propagation and Representation Refinement 16

B.2 Adaptive Cluster Discovery via DP–Diffusion Coupling 19

C Implementation Details of DD-FSC **23**

C.1 Training and Inference Algorithm 23

C.2 Reproducibility Protocol 24

C.3 Computational Complexity and Resource Footprint 27

D Additional Results on Image Clustering Benchmarks **29**

E Ablation Protocols and Cross-Benchmark Analysis **30**

E.1 Unknown-K Ablation Variants 31

E.2 Unknown-K Ablation Variants 31

E.3 Additional benchmark ablations 32

F Run-to-Run Variability and Statistical Significance **34**

G Robustness Analysis **36**

G.1 OpenCLIP Backbone Variants 36

G.2 Few-Shot Supervision Conditions 37

H Additional Hyperparameter Sensitivity Analysis **38**

H.1 Score-Guided Exploration Controls 39

H.2 Representation Objective Weights 39

H.3 Adaptive Cluster Discovery Controls 41

I Visualization Analysis **42**

I.1 Comparison with Representative Methods 42

I.2 Evolution of Representation Structure 43

APPENDIX

A BENCHMARK DATASETS

We evaluate DD-FSC on image benchmarks spanning large-scale object recognition, handwritten digits, fashion images, fine-grained recognition, and plant identification. Table 6 summarizes the number of classes and the training and held-out split sizes for the datasets and subsets considered in our experiments. We list ImageNet-100 separately for few-shot clustering and generalized category discovery (GCD), as the two evaluations use different class subsets.

Table 6: Dataset and subset statistics. Train Size includes both labeled and unlabeled training images. Val./Test Size denotes the corresponding held-out split.

Dataset	Num Classes	Train Size	Val./Test Size
Main few-shot clustering benchmarks			
ImageNet-50	50	65,000	2,500
ImageNet-100	100	130,000	5,000
ImageNet-200	200	260,000	10,000
ImageNet-1K	1,000	1,281,167	50,000
Additional few-shot clustering benchmarks			
MNIST	10	60,000	10,000
Fashion-MNIST	10	60,000	10,000
CIFAR-10	10	50,000	10,000
CIFAR-20	20	50,000	10,000
STL-10	10	5,000	8,000
ImageNet-Dogs	15	15,600	3,900
Generalized category discovery benchmarks			
CIFAR-100	100	50,000	10,000
ImageNet-100 (GCD)	100	127,115	5,000
CUB-200-2011	200	5,994	5,794
Stanford Cars	196	8,144	8,041
Herbarium19	683	34,225	9,565

Few-shot clustering benchmarks. The main evaluation uses ImageNet-50, ImageNet-100, ImageNet-200, and ImageNet-1K (Deng et al., 2009). The first three are the balanced subsets used in our few-shot clustering experiments, containing 50, 100, and 200 categories, respectively, with 1,300 training and 50 validation images per category. These subset sizes describe our selected categories rather than a uniform per-category training size for ImageNet-1K.

The additional standard-scale evaluation uses MNIST (Deng, 2012), Fashion-MNIST (Xiao et al., 2017), CIFAR-10 and CIFAR-20 (Krizhevsky, 2009), STL-10 (Coates et al., 2011), and ImageNet-Dogs (Chang et al., 2017). CIFAR-20 uses the 20 coarse superclasses of CIFAR-100. ImageNet-Dogs contains 19,500 images from 15 dog categories drawn from ImageNet. The table reports the 15,600/3,900 training and test partition used in our experiments, corresponding to an 80/20 split of this subset.

Generalized category discovery benchmarks. The GCD evaluation uses CIFAR-100 (Krizhevsky, 2009), ImageNet-100 (Deng et al., 2009), CUB-200-2011 (Wah et al., 2011), Stanford Cars (Krause et al., 2013), and Herbarium19 (Tan et al., 2019) under both known- and unknown-K settings. We follow the standard category splits used in prior GCD work (Vaze et al., 2022). CIFAR-100 contains 80 Old and 20 New categories. ImageNet-100 contains 50 Old and 50 New categories, CUB-200-2011 contains 100 Old and 100 New categories, and Stanford Cars contains 98 categories in each set. Herbarium19 contains 341 Old and 342 New categories.

In the GCD evaluation, ImageNet-100 denotes the 100-class subset generated with the default seed of 0 in the released GCD implementation (Vaze et al., 2022). This subset is distinct from the balanced ImageNet-100 subset used in our few-shot clustering experiments. It includes all original ImageNet training images from the selected classes, yielding 127,115 training images; the corresponding ImageNet validation subset contains 5,000 images.

For each GCD benchmark, we randomly select 50% of the training images belonging to Old categories to form the labeled set. The unlabeled set contains the remaining training images from Old categories and all training images from New categories. Under this protocol, ImageNet-100 contains 31,860 labeled training images and 95,255 unlabeled training images.

B Additional Methodological and Theoretical Details

This section provides additional details for the three core modules of DD-FSC. Score-guided semantic propagation and density-guided ambiguity resolution share the same score model and feature-space trajectories, so we present them jointly. We then detail adaptive cluster discovery through DP–diffusion coupling, including its variational formulation and coordinate updates.

B.1 Score-Guided Semantic Propagation and Representation Refinement

This subsection proves Theorem 1 and provides the theoretical details omitted from Section 3.2. We first distinguish the conditional Gaussian-corruption score from the noisy marginal score. We then connect denoising score matching to finite score-estimation error and derive the corresponding density-ascent guarantee. Throughout this subsection, $x \in \mathbb{R}^d$ denotes an adapted feature and corresponds to h in the main text.

B.1.1 Gaussian smoothing and the noisy marginal score

Let μ_{data} denote the probability distribution of the clean adapted features. This measure may be supported on the unit sphere because the features are ℓ_2-normalized before score modeling. At noise level $\sigma > 0$, let $\mathbf{I}$ be the $d \times d$ identity matrix. The Gaussian corruption kernel is

$$q_\sigma(\widetilde{x} \mid x) = \mathcal{N}(\widetilde{x}; x, \sigma^2 \mathbf{I}). \tag{B.1.1}$$

The corresponding noisy marginal density is

$$p_\sigma(\widetilde{x}) = \int q_\sigma(\widetilde{x} \mid x)\, \mathrm{d}\mu_{\mathrm{data}}(x). \tag{B.1.2}$$

For every $\sigma > 0$, Gaussian convolution makes $p_\sigma(\widetilde{x})$ strictly positive and infinitely differentiable, even when μ_{data} does not admit an ambient Euclidean density. Its log-density gradient is therefore well-defined. The conditional Gaussian score is

$$\nabla_{\widetilde{x}} \log q_\sigma(\widetilde{x} \mid x) = \frac{x - \widetilde{x}}{\sigma^2}. \tag{B.1.3}$$

Differentiating equation B.1.2 under the integral and normalizing by $p_\sigma(\widetilde{x})$ gives

$$\nabla_{\widetilde{x}} \log p_\sigma(\widetilde{x}) = \frac{1}{p_\sigma(\widetilde{x})} \int \nabla_{\widetilde{x}} q_\sigma(\widetilde{x} \mid x)\, \mathrm{d}\mu_{\mathrm{data}}(x) = \mathbb{E}\left[\frac{x - \widetilde{x}}{\sigma^2} \,\middle|\, \widetilde{x} \right]. \tag{B.1.4}$$

Equation B.1.4 is the denoising score identity. It shows that the marginal score points from a noisy feature toward the posterior mean of its clean counterpart. The noise-conditional score network is therefore trained to approximate

$$s_\psi(\widetilde{x}, \sigma) \simeq \nabla_{\widetilde{x}} \log p_\sigma(\widetilde{x}), \tag{B.1.5}$$

rather than the score of an individual Gaussian corruption kernel. Let $\mathcal{S}$ be the noise schedule, $p_{\mathcal{S}}$ its training distribution, and $\lambda(\sigma)$ the noise-level weight. The corresponding denoising score-matching objective is

$$\mathcal{L}_{\text{score}}(\psi) = \mathbb{E}_{\substack{\sigma\sim p_S,\, x\sim\mu_{\text{data}},\\ \widetilde{x}\sim q_\sigma(\cdot|x)}} \left[\lambda(\sigma) \left\| s_\psi(\widetilde{x},\sigma) - \frac{x-\widetilde{x}}{\sigma^2} \right\|_2^2 \right], \tag{B.1.6}$$

Here, $\lambda(\sigma)$ balances the contributions of different noise levels.

For a fixed noise level, define the conditional score-matching risk $\mathcal{R}_\sigma(s)$ by removing the expectation over σ and the scale-dependent weight from equation B.1.6. Let $g_\sigma(\widetilde{x}) = \nabla_{\widetilde{x}} \log p_\sigma(\widetilde{x})$. The conditional-expectation identity in equation B.1.4 yields the exact decomposition

$$\mathcal{R}_\sigma(s) - \mathcal{R}_\sigma(g_\sigma) = \mathbb{E}_{\widetilde{x}\sim p_\sigma} \left[\|s(\widetilde{x},\sigma) - g_\sigma(\widetilde{x})\|_2^2 \right]. \tag{B.1.7}$$

Thus, the excess denoising score-matching risk is exactly the distribution-averaged squared score-estimation error. This identity connects the score-training objective directly to the approximation error appearing in Theorem 1. It does not, by itself, imply a uniform pointwise error bound.

B.1.2 Finite-error density-ascent guarantee

Fix an encoder state and a noise level $\sigma > 0$. Define $\ell_\sigma(x) = \log p_\sigma(x)$, $g_\sigma(x) = \nabla \ell_\sigma(x)$, and $e_\sigma(x) = s_\psi(x,\sigma) - g_\sigma(x)$. Let $\mathcal{U}$ be a compact convex neighborhood containing the deterministic update segment. Since p_σ is strictly positive and smooth, g_σ is locally Lipschitz; hence there exists a finite $L_\sigma > 0$ such that g_σ is L_σ-Lipschitz on $\mathcal{U}$. We further assume that the learned score has bounded local error:

$$\sup_{x\in\mathcal{U}} \|e_\sigma(x)\|_2 \le \delta_\sigma. \tag{B.1.8}$$

For a fixed noise level, let $\epsilon > 0$ be the step size, $\gamma > 0$ the stochastic exploration magnitude, and $\xi^{(m)} \sim \mathcal{N}(0,\mathbf{I})$. We analyze the implemented score-guided update in equation 1, writing its feature state as $\widetilde{x}^{(m)}$ to match the notation used in this proof.

Proof of Theorem 1. Conditioning on $\widetilde{x}^{(m)} = x$ and using $\mathbb{E}[\xi^{(m)}] = 0$ gives

$$\mathbb{E}\left[\widetilde{x}^{(m+1)} - \widetilde{x}^{(m)} \,\middle|\, \widetilde{x}^{(m)} = x \right] = \epsilon s_\psi(x,\sigma). \tag{B.1.9}$$

Combining equation B.1.9 with the error bound in equation B.1.8 yields

$$\left\| \mathbb{E}\left[\widetilde{x}^{(m+1)} - \widetilde{x}^{(m)} \,\middle|\, \widetilde{x}^{(m)} = x \right] - \epsilon g_\sigma(x) \right\|_2 \le \epsilon\delta_\sigma. \tag{B.1.10}$$

Let $x_{\text{d}}^+ = x + \epsilon s_\psi(x,\sigma)$ denote the deterministic endpoint. The L_σ-smoothness of ℓ_σ, the decomposition $s_\psi = g_\sigma + e_\sigma$, and the Cauchy–Schwarz inequality give

$$\begin{aligned}
\ell_\sigma(x_{\text{d}}^+) - \ell_\sigma(x) &\ge \epsilon g_\sigma(x)^\top s_\psi(x,\sigma) - \frac{L_\sigma\epsilon^2}{2} \|s_\psi(x,\sigma)\|_2^2 \\
&= \epsilon \|g_\sigma(x)\|_2^2 + \epsilon g_\sigma(x)^\top e_\sigma(x) - \frac{L_\sigma\epsilon^2}{2} \|s_\psi(x,\sigma)\|_2^2 \\
&\ge \epsilon \|g_\sigma(x)\|_2 \left(\|g_\sigma(x)\|_2 - \delta_\sigma \right) - \frac{L_\sigma\epsilon^2}{2} \|s_\psi(x,\sigma)\|_2^2 .
\end{aligned} \tag{B.1.11}$$

If $\delta_\sigma < \|g_\sigma(x)\|_2$, the lower bound in equation B.1.11 is strictly positive whenever

$$0 < \epsilon < \frac{2\|g_\sigma(x)\|_2 \big(\|g_\sigma(x)\|_2 - \delta_\sigma \big)}{L_\sigma \|s_\psi(x,\sigma)\|_2^2}. \tag{B.1.12}$$

The same error assumption also gives the following directional alignment guarantee:

$$\frac{g_\sigma(x)^\top s_\psi(x,\sigma)}{\|g_\sigma(x)\|_2\,\|s_\psi(x,\sigma)\|_2} \geq \frac{\|g_\sigma(x)\|_2 - \delta_\sigma}{\|g_\sigma(x)\|_2 + \delta_\sigma} > 0. \tag{B.1.13}$$

Therefore, when the true smoothed-density gradient dominates the local score-estimation error, the learned deterministic drift forms an acute angle with that gradient. Under the step-size condition in equation B.1.12, it also produces a strict increase in the smoothed log-density. The stochastic term has zero conditional mean and therefore does not change this drift guarantee. However, it prevents individual sampled trajectories from being interpreted as monotone deterministic optimizers. □

The analysis above concerns the Gaussian-smoothed feature density p_σ. This distinction is important because the clean adapted features are ℓ_2-normalized and may be supported on the unit sphere, in which case an ambient clean-feature density need not exist. Gaussian smoothing produces a positive and smooth density for every $\sigma > 0$, so the finite-error guarantee remains well-defined without assuming a full-dimensional clean distribution.

Theorem 1 is a geometric guarantee and does not claim that a score-guided trajectory preserves category identity. DD-FSC imposes semantic constraints separately. Trajectories initialized from labeled supports define density-informed category anchors, and only high-confidence unlabeled candidates are retained through the selection rule in equation B.1.17. The theorem therefore justifies the density-directed exploration step, while few-shot anchors, confidence filtering, and the contrastive objectives control semantic consistency.

B.1.3 Score-Guided Semantic Expansion

Let $h_i = f_\phi(x_i)$ denote the encoder feature of sample x_i, and let $\sigma_L > \cdots > \sigma_1 > 0$ be the noise schedule. At level ℓ, use M_ℓ steps of size $\epsilon_\ell > 0$ and initialize $\tilde{h}_{i,\ell}^{(0)}$ with the terminal state of the preceding, noisier level (and initialize the first level from h_i plus the corresponding Gaussian perturbation). The feature is updated for M_ℓ steps by

$$\tilde{h}_{i,\ell}^{(m+1)} = \tilde{h}_{i,\ell}^{(m)} + \epsilon_\ell s_\psi\left(\tilde{h}_{i,\ell}^{(m)}, \sigma_\ell\right) + \gamma\sqrt{2\epsilon_\ell}\,\xi_{i,\ell}^{(m)}, \quad \xi_{i,\ell}^{(m)} \sim \mathcal{N}(0, \mathbf{I}). \tag{B.1.14}$$

Equation equation B.1.14 is the score-only base trajectory and is recovered from equation 6 by setting $\lambda_{\mathrm{DP}} = 0$. In the complete coupled model, equation 6 is used at every annealed update for trajectories initialized from labeled supports and ordinary mini-batch samples, with $t_i^{(m)} = \tilde{h}_{i,\ell}^{(m)}$, $\eta_m = \epsilon_\ell$, and $\sigma_m = \sigma_\ell$. The same coupled update is applied to posterior-weighted prototypes as specified in Appendix B.2. When DP–diffusion coupling is disabled, all three trajectory types reduce to equation B.1.14.

The final guided feature produced by the applicable update is

$$\hat{h}_i = \tilde{h}_{i,1}^{(M_1)}. \tag{B.1.15}$$

All quantities in these updates are in feature space; the procedure does not synthesize a new image. The gradient flow associated with each downstream objective is specified in the corresponding subsections below.

For each observed category $y \in \mathcal{Y}_l$, let $\mathcal{D}_l^y = \{x_i : (x_i, y_i) \in \mathcal{D}_l,\ y_i = y\}$. Using the terminal Langevin features defined in equation B.1.15, we construct a density-guided category anchor:

$$a_y = \frac{\dfrac{1}{|\mathcal{D}_l^y|}\displaystyle\sum_{x_i \in \mathcal{D}_l^y} \hat{h}_i}{\left\| \dfrac{1}{|\mathcal{D}_l^y|}\displaystyle\sum_{x_i \in \mathcal{D}_l^y} \hat{h}_i \right\|_2}. \tag{B.1.16}$$

Thus, a_y summarizes the density-guided endpoints of category y's few-shot samples.

For each unlabeled sample x_j, we compute its normalized feature $\bar{h}_j = h_j/\|h_j\|_2$ and evaluate its assignment scores using equation 2. Let

$$\mathcal{I}_y = \left\{ j : \arg\max_{c \in \mathcal{Y}_l} r_{jc} = y,\ \max_{c \in \mathcal{Y}_l} r_{jc} \geq \delta \right\} \tag{B.1.17}$$

denote the indices of unlabeled samples assigned to category y with sufficient confidence. We retain the indices of the at most P largest scores r_{jy} within $\mathcal{I}_y$, denoted by $\mathrm{TopP}(\mathcal{I}_y; r_{jy})$. The selected candidates and expanded semantic set are

$$\mathcal{C}_y = \{x_j : j \in \mathrm{TopP}(\mathcal{I}_y; r_{jy})\}, \qquad \widehat{\mathcal{D}}_y = \mathcal{D}_l^y \cup \mathcal{C}_y. \tag{B.1.18}$$

Samples outside these selected sets retain their unlabeled status for subsequent clustering. The full-data FCL objective is defined in equation 3. During optimization, we use its mini-batch estimator and omit an anchor if the batch contains no other member of the same expanded semantic set. The resulting objective increases within-set similarity relative to all other samples in the batch.

B.1.4 Density-Guided Pair Alignment and Gradient Flow

Unlike category-anchor construction, which initializes trajectories only from labeled supports, DGCL applies the annealed sampler to every feature in the current mini-batch. For each $x_i \in \mathcal{B}$, the update in equation B.1.14, initialized from $h_i = f_\phi(x_i)$, produces the density-guided endpoint $\hat{h}_i$. This batch-wide construction also covers samples outside every expanded semantic set, including those from unobserved categories. It transfers diffusion-derived density information without assigning semantic labels or converting uncertain predictions into hard pseudo-labels.

Before evaluating the DGCL objective in equation 4, $\hat{h}_i$ is detached from the sampling trajectory and therefore acts as a fixed target within that neural update. Let $J_\phi(x_i) = \partial h_i/\partial\phi$ denote the encoder Jacobian. The corresponding gradient path is

$$\nabla_\phi \mathcal{L}_{\mathrm{DGCL}} = \frac{2}{|\mathcal{B}|} \sum_{x_i \in \mathcal{B}} J_\phi(x_i)^\top \left(h_i - \mathrm{sg}(\hat{h}_i) \right), \qquad \nabla_\psi \mathcal{L}_{\mathrm{DGCL}} = 0. \tag{B.1.19}$$

Thus, DGCL updates the encoder through the original-feature branch but does not backpropagate through the trajectory or use its alignment error to alter the score field. The score network is optimized separately by $\mathcal{L}_{\mathrm{score}}$, and new endpoints are recomputed from the current features during subsequent mini-batch updates.

DGCL is deliberately class-agnostic and contains no explicit negatives. Its role is to move each feature toward a density-guided counterpart, not to assert category identity or guarantee monotone density increase along every stochastic trajectory. FCL supplies semantic attraction within the confidence-filtered expanded sets, while PCL supplies prototype-level separation between active components. DGCL therefore extends density-aware refinement to samples beyond the expanded sets without replacing the semantic and cluster-level constraints.

B.2 Adaptive Cluster Discovery via DP–Diffusion Coupling

This subsection specifies the local responsibilities, mini-batch prototype construction, and truncated variational inference underlying the coupling introduced in Section 3.4. The guidance field, coupled trajectory, PCL objective, and complete training objective remain as defined in equation 5, equation 6, equation 7, and equation 8, respectively.

B.2.1 DP-Guided Trajectories and Prototype Construction

Let Q denote the current variational posterior, $\mathcal{A}$ the active-component set, π_k the mixing weight of component k, and $\Theta = \{(\mu_k, \Sigma_k)\}_{k=1}^{K_{\max}}$ its component parameters. Let $t_i^{(m)}$ denote the trajectory state of sample i after m updates. We evaluate the local responsibility

$$\rho_{ik}^{(m)} = \frac{\exp\Big(\mathbb{E}_Q[\log \pi_k] + \mathbb{E}_Q[\log \mathcal{N}(t_i^{(m)}; \mu_k, \Sigma_k)] \Big)}{\sum_{j \in \mathcal{A}} \exp\Big(\mathbb{E}_Q[\log \pi_j] + \mathbb{E}_Q[\log \mathcal{N}(t_i^{(m)}; \mu_j, \Sigma_j)] \Big)}. \tag{B.2.1}$$

This quantity is evaluated from the current variational DP posterior. It is not a conditional factor in the mean-field family. Substituting $\rho_{ik}^{(m)}$ into the posterior-weighted field in equation 5 makes the guidance location-dependent. The bounded direction in equation 6 preserves this direction whenever the raw field is nonzero while limiting its norm to one. The resulting trajectory constructs guided representations; it is not included as an observed-data factor in the DP mixture likelihood.

For a mini-batch $\mathcal{B}$, let z_i be the latent component assignment and $\varepsilon_{\mathcal{B}} > 0$ the batch-mass threshold. Define

$$\mathcal{A}_{\mathcal{B}} = \left\{ k \in \mathcal{A} : \sum_{x_i \in \mathcal{B}} Q(z_i = k) \geq \varepsilon_{\mathcal{B}} \right\}. \tag{B.2.2}$$

For each $k \in \mathcal{A}_{\mathcal{B}}$, the normalized posterior-weighted prototype is

$$\mathbf{c}_k = \frac{\sum_{x_i \in \mathcal{B}} Q(z_i = k) h_i}{\left\| \sum_{x_i \in \mathcal{B}} Q(z_i = k) h_i \right\|_2}. \tag{B.2.3}$$

Starting from $\mathbf{c}_k$, the update in equation 6 produces a guided endpoint. We renormalize this endpoint to obtain $\mathbf{c}'_k$ and use it as a stop-gradient target. These prototypes enter the PCL objective in equation 7. When $|\mathcal{A}_{\mathcal{B}}| < 2$, we set $\mathcal{L}_{\mathrm{PCL}} = 0$. We next detail the variational DP terms entering the complete objective in equation 8.

B.2.2 Truncated DP Mixture Specification

Let $\alpha > 0$ be the DP concentration parameter, G_0 the base measure, and ϑ_k the parameter of component k. The infinite stick-breaking construction is

$$v_k \sim \mathrm{Beta}(1, \alpha), \qquad \pi_k = v_k \prod_{j<k} (1 - v_j), \qquad \vartheta_k \sim G_0. \tag{B.2.4}$$

For computation, we truncate the representation at $K_{\max}$ and set $v_{K_{\max}} = 1$. Therefore, $\sum_{k=1}^{K_{\max}} \pi_k = 1$. The truncation controls computational cost and does not specify the effective number of active clusters.

For each component k, we use a diagonal Gaussian likelihood

$$p(h_i \mid z_i = k, \vartheta_k) = \mathcal{N}(h_i; \mu_k, \Sigma_k), \qquad \Sigma_k = \mathrm{diag}\left(\lambda_{k1}^{-1}, \ldots, \lambda_{kd_h}^{-1}\right), \tag{B.2.5}$$

where $\boldsymbol{\lambda}_k = (\lambda_{k1}, \ldots, \lambda_{kd_h})$ contains the component-wise precisions.

We adopt a factorized Normal-Gamma base measure:

$$p(\vartheta_k \mid G_0) = \prod_{d=1}^{d_h} \mathcal{N}\left(\mu_{kd}; m_{0d}, (\beta_0 \lambda_{kd})^{-1}\right) \mathrm{Gamma}\left(\lambda_{kd}; a_0, b_{0d}\right). \tag{B.2.6}$$

Here, the Gamma distribution uses the shape-rate parameterization, $m_0 \in \mathbb{R}^{d_h}$ is the prior mean, and $\beta_0 > 0$, $a_0 > 0$, and $b_{0d} > 0$ are prior hyperparameters. This diagonal construction avoids estimating and inverting a full covariance matrix in the high-dimensional feature space.

B.2.3 Factorization and evidence lower bound

Let $H = \{h_i\}_{i=1}^N$, $Z = \{z_i\}_{i=1}^N$, $V = \{v_k\}_{k=1}^{K_{\max}-1}$, and $\Theta = \{\vartheta_k\}_{k=1}^{K_{\max}}$. The truncated DP mixture factorizes as

$$p(H, Z, V, \Theta) = p(V \mid \alpha) p(\Theta \mid G_0) \prod_{i=1}^{N} p(z_i \mid V) p(h_i \mid z_i, \Theta). \tag{B.2.7}$$

Here, $p(V \mid \alpha) = \prod_{k=1}^{K_{\max}-1} \mathrm{Beta}(v_k; 1, \alpha)$ and $p(\Theta \mid G_0) = \prod_{k=1}^{K_{\max}} p(\vartheta_k \mid G_0)$. The final stick $v_{K_{\max}} = 1$ is deterministic. We use the mean-field family

$$Q(V, \Theta, Z) = \left[\prod_{k=1}^{K_{\max}-1} Q(v_k) \right] \left[\prod_{k=1}^{K_{\max}} Q(\vartheta_k) \right] \left[\prod_{i=1}^{N} Q(z_i) \right]. \tag{B.2.8}$$

For compactness, we denote the three bracketed products by $Q(V)$, $Q(\Theta)$, and $Q(Z)$, respectively. The final stick $v_{K_{\max}}$ is fixed to one and therefore has no variational factor. The evidence lower bound is

$$\begin{aligned}\mathcal{L}_{\mathrm{ELBO}} = \mathbb{E}_Q[\log p(V \mid \alpha) - \log Q(V)] + \mathbb{E}_Q[\log p(\Theta \mid G_0) - \log Q(\Theta)] \\ + \sum_{i=1}^{N} \mathbb{E}_Q[\log p(z_i \mid V) + \log p(h_i \mid z_i, \Theta)] - \mathbb{E}_Q[\log Q(Z)].\end{aligned} \tag{B.2.9}$$

For variational coordinate updates, we maximize the full ELBO in equation B.2.9. For joint neural optimization, we use its normalized form

$$\overline{\mathcal{L}}_{\mathrm{ELBO}} = \frac{1}{N d_h} \mathcal{L}_{\mathrm{ELBO}}. \tag{B.2.10}$$

Multiplication by the positive constant $(N d_h)^{-1}$ does not change the variational optimum. It prevents the ELBO contribution from growing with the dataset size or feature dimension.

B.2.4 Coordinate-Ascent Variational Updates

Cluster assignments. Let $r_{ik} = Q(z_i = k)$. The coordinate update is

$$r_{ik} = \frac{\exp(\ell_{ik})}{\sum_{j=1}^{K_{\max}} \exp(\ell_{ij})}, \tag{B.2.11}$$

where

$$\ell_{ik} = \mathbb{E}_Q[\log \pi_k] + \mathbb{E}_Q[\log p(h_i \mid \vartheta_k)]. \tag{B.2.12}$$

These variational responsibilities are used for posterior-weighted prototype construction. During trajectory refinement, analogous location-dependent responsibilities are recomputed at each trajectory state, as defined in Section 3.4.

Stick variables. For each $k < K_{\max}$, let $N_k = \sum_{i=1}^{N} r_{ik}$ denote the effective posterior count.

$$Q(v_k) = \mathrm{Beta}(\widetilde{a}_k, \widetilde{b}_k), \qquad \widetilde{a}_k = 1 + N_k, \qquad \widetilde{b}_k = \alpha + \sum_{j=k+1}^{K_{\max}} N_j. \tag{B.2.13}$$

Let $\mathrm{digamma}(a) = \frac{\mathrm{d}}{\mathrm{d}a} \log \Gamma(a)$ denote the digamma function. The expected log mixing weight is

$$\begin{aligned}\mathbb{E}_Q[\log \pi_k] = \ & \mathrm{digamma}(\widetilde{a}_k) - \mathrm{digamma}(\widetilde{a}_k + \widetilde{b}_k) \\ & + \sum_{j<k} \left[\mathrm{digamma}(\widetilde{b}_j) - \mathrm{digamma}(\widetilde{a}_j + \widetilde{b}_j)\right], \qquad k < K_{\max}.\end{aligned} \tag{B.2.14}$$

For the final component,

$$\mathbb{E}_Q[\log \pi_{K_{\max}}] = \sum_{j<K_{\max}} \left[\mathrm{digamma}(\widetilde{b}_j) - \mathrm{digamma}(\widetilde{a}_j + \widetilde{b}_j)\right]. \tag{B.2.15}$$

The posterior mean mixing weights used for active-component selection are

$$\mathbb{E}_Q[\pi_k] = \begin{cases} \dfrac{\widetilde{a}_k}{\widetilde{a}_k + \widetilde{b}_k} \displaystyle\prod_{j<k} \frac{\widetilde{b}_j}{\widetilde{a}_j + \widetilde{b}_j}, & k < K_{\max}, \\ \displaystyle\prod_{j<K_{\max}} \frac{\widetilde{b}_j}{\widetilde{a}_j + \widetilde{b}_j}, & k = K_{\max}. \end{cases} \tag{B.2.16}$$

The expression follows from the independence of the variational stick factors.

Cluster parameters. Let $r_{ik} = Q(z_i = k)$ and $N_k = \sum_{i=1}^{N} r_{ik}$. For $N_k > 0$, define the responsibility-weighted sufficient statistics

$$\bar{h}_{kd} = \frac{1}{N_k} \sum_{i=1}^{N} r_{ik} h_{id}, \qquad S_{kd} = \sum_{i=1}^{N} r_{ik} \left(h_{id} - \bar{h}_{kd}\right)^2. \tag{B.2.17}$$

Under the Normal-Gamma base measure in equation B.2.6, the variational factor for component k has the conjugate form

$$Q(\vartheta_k) = \prod_{d=1}^{d_h} \mathcal{N}\left(\mu_{kd}; m_{kd}, (\beta_k \lambda_{kd})^{-1}\right) \mathrm{Gamma}\left(\lambda_{kd}; a_k, b_{kd}\right). \tag{B.2.18}$$

Its parameters are updated as

$$\beta_k = \beta_0 + N_k, \tag{B.2.19a}$$

$$m_{kd} = \frac{\beta_0 m_{0d} + N_k \bar{h}_{kd}}{\beta_k}, \tag{B.2.19b}$$

$$a_k = a_0 + \frac{N_k}{2}, \tag{B.2.19c}$$

$$b_{kd} = b_{0d} + \frac{1}{2}\left[S_{kd} + \frac{\beta_0 N_k}{\beta_k}\left(\bar{h}_{kd} - m_{0d}\right)^2\right]. \tag{B.2.19d}$$

When $N_k = 0$, the variational factor reduces to its prior.

The expected component log likelihood required by the assignment update is

$$\mathbb{E}_Q[\log p(h_i \mid \vartheta_k)] = \frac{1}{2} \sum_{d=1}^{d_h} \left[\mathrm{digamma}(a_k) - \log(2\pi b_{kd}) - \frac{a_k (h_{id} - m_{kd})^2}{b_{kd}} - \frac{1}{\beta_k}\right]. \tag{B.2.20}$$

For a generic trajectory state $t \in \mathbb{R}^{d_h}$, let $m_k = (m_{k1}, \ldots, m_{kd_h})$. The expectation used by the DP guidance field has the closed form

$$\mathbb{E}_Q\left[\Sigma_k^{-1}(\mu_k - t)\right] = \mathrm{diag}\left(\frac{a_k}{b_{k1}}, \ldots, \frac{a_k}{b_{kd_h}}\right)(m_k - t). \tag{B.2.21}$$

Thus, both the assignment scores and the DP guidance field can be computed directly from the variational parameters.

B.2.5 Active components and cluster-count inference

In the known-K setting, we set $K_{\max} = K$ and retain all components:

$$\mathcal{A} = \{1, \ldots, K\}. \tag{B.2.22}$$

In the unknown-K setting, the active set is determined by the posterior mixing weights:

$$\mathcal{A} = \{k : \mathbb{E}_Q[\pi_k] \geq \varepsilon\}. \tag{B.2.23}$$

If the thresholded set is empty, we retain the component with the largest posterior mean weight:

$$\mathcal{A} = \left\{\arg\max_k \mathbb{E}_Q[\pi_k]\right\}.$$

This safeguard ensures that the local responsibilities and the DP guidance field remain well defined. A component becomes active when its posterior mass reaches the threshold and becomes inactive when its mass falls below it. The inferred number of clusters is

$$\widehat{K} = |\mathcal{A}|. \tag{B.2.24}$$

Thus, $K_{\max}$ is a computational truncation ceiling rather than the predicted category count. The estimate $\widehat{K}$ is obtained from the posterior-mass active set in equation B.2.23, together with the nonempty-set safeguard above.

C Implementation Details of DD-FSC

This section describes the training and inference procedure, implementation settings required for reproducibility, and computational cost of DD-FSC.

C.1 Training and Inference Algorithm

Algorithm 1 summarizes the optimization and final inference procedures. In the known-K setting, we set $K_{\max} = K$ and retain all components. In the unknown-K setting, $K_{\max}$ is a truncation level, and the active set is determined from the posterior mixing weights using equation B.2.23. Components enter or leave the active set as their posterior masses change.

Algorithm 1 Training and inference of DD-FSC

Require: Data set $\mathcal{D} = \mathcal{D}_l \cup \mathcal{D}_u$, encoder f_ϕ, score network s_ψ, DP truncation $K_{\max}$, known- or unknown-K setting, and optimization hyperparameters

Initialize ϕ, ψ, and the variational factors $q(Z)$, $q(V)$, and $q(\Theta)$
for each outer training iteration **do**
  Compute $H = \{f_\phi(x_i) : x_i \in \mathcal{D}\}$ and, with ϕ, ψ fixed, update $q(Z)$, $q(V)$, and $q(\Theta)$ using equations B.2.11– B.2.19
  Set $\mathcal{A} = \{1, \dots, K\}$ when K is known; otherwise determine $\mathcal{A}$ from $\mathbb{E}_q[\pi_k]$ using equation B.2.16 and equation B.2.23, including its nonempty-set safeguard
  if $\mathcal{D}_l \neq \varnothing$ **then**
    Construct support endpoints using equation 6, with $\lambda_{\mathrm{DP}} = 0$ reducing it to equation B.1.14; construct category anchors and semantic sets using equations B.1.16 and B.1.18
  else
    Omit label-derived semantic expansion and set $\mathcal{L}_{\mathrm{FCL}} = 0$
  end if
  for each neural mini-batch $\mathcal{B}$ **do**
    Compute the current features and their guided counterparts using equation 6, with $\lambda_{\mathrm{DP}} = 0$ reducing it to equation B.1.14; use the detached counterparts as targets for $\mathcal{L}_{\mathrm{DGCL}}$
    Form $\mathcal{A}_\mathcal{B}$ and its posterior-weighted prototypes using equations B.2.2 and B.2.3
    Obtain the guided prototype targets using the same update in equation 6; compute $\mathcal{L}_{\mathrm{PCL}}$ using equation 7 when $|\mathcal{A}_\mathcal{B}| \geq 2$, and otherwise set $\mathcal{L}_{\mathrm{PCL}} = 0$
    Evaluate $\widehat{\mathcal{L}}_{\mathrm{ELBO}}$ using equation C.1, and then compute $\mathcal{L}_{\mathrm{total}}$ in equation 8 with the variational factors fixed
    Update ϕ through the encoder-dependent terms and update ψ through $\mathcal{L}_{\mathrm{score}}$
  end for
end for
Recompute the final feature collection $H = \{f_\phi(x_i) : x_i \in \mathcal{D}\}$ with ϕ fixed
if K is known **then**
  if $\mathcal{D}_l \neq \varnothing$ and $\lambda_{\mathrm{FCL}} > 0$ **then**
    Run support-anchored spherical k-means with K clusters on H using 30 iterations, three seed-deterministic restarts, and support-anchor weight 100
  else
    Run standard k-means with K clusters on the ℓ_2-normalized features using 20 initializations and at most 300 iterations (DD-FSC-U and known-K *w/o FCL*)
  end if
  Set $\{\hat{z}_i\}_{i=1}^N$ to the assignments returned by the corresponding known-K procedure
else
  Refresh $q(Z)$, $q(V)$, and $q(\Theta)$ for ten CAVI sweeps with ϕ and ψ fixed
  Compute $\mathbb{E}_q[\pi_k]$, determine $\mathcal{A}$ using equation B.2.23, and set $\hat{z}_i = \arg\max_{k \in \mathcal{A}} q(z_i = k)$
end if

Ensure: Trained parameters ϕ and ψ, cluster assignments $\{\hat{z}_i\}_{i=1}^N$, and, when K is unknown, $\widehat{K} = |\mathcal{A}|$ as defined in equation B.2.24

For neural optimization, we estimate the normalized ELBO in equation B.2.10 using uniformly sampled mini-batches. Let $\mathcal{G}(Q)$ collect its global stick and component prior–entropy terms. For a

mini-batch $\mathcal{B}$, the estimator is

$$\widehat{\mathcal{L}}_{\mathrm{ELBO}} = \frac{\mathcal{G}(Q)}{Nd_h} + \frac{1}{|\mathcal{B}|d_h} \sum_{i\in\mathcal{B}} \sum_{k=1}^{K_{\max}} r_{ik} \left(\mathbb{E}_Q[\log \pi_k] + \mathbb{E}_Q[\log p(h_i \mid \vartheta_k)] - \log r_{ik} \right). \tag{C.1}$$

The first term contains the global variational contribution, whereas the second is the normalized mini-batch estimate of the local assignment and likelihood terms.

During each neural phase, the variational factors are treated as constants, while $\widehat{\mathcal{L}}_{\mathrm{ELBO}}$ remains differentiable through the current encoder features. The procedure therefore does not differentiate through the coordinate-ascent updates. As defined in Appendix B.1, $\mathcal{L}_{\mathrm{score}}$ updates ψ, whereas the remaining encoder-dependent terms update ϕ. The final posterior refresh ensures that the reported assignments and active set correspond to the learned representation.

C.2 Reproducibility Protocol

This subsection specifies the model architecture, feature construction, optimization, score-guided sampling, semantic expansion, and variational DP inference used in the main experiments. Together with Algorithm 1, these specifications define the reported DD-FSC protocol without requiring readers to reconstruct its implementation from source code. Within each dataset and setting, all configurations are fixed across the three reported seeds unless stated otherwise. The main-table protocol uses paired support-split and neural-optimization seeds 1, 2, and 3 for known-K, and 42, 43, and 44 for unknown-K. The final-epoch checkpoint is evaluated without label-based checkpoint selection.

Backbone, adapter, and feature views. All main experiments use a frozen OpenCLIP ViT-B/16 image encoder with the publicly released `laion2b_s34b_b88k` checkpoint. Its 512-dimensional image embedding is ℓ_2-normalized before entering the adapter. The adapted representation $h = f_\phi(x)$ is normalized again before score modeling, DP inference, and contrastive learning. No OpenCLIP parameter is updated.

The adapter consists of two residual MLP blocks. Each block applies LayerNorm, a linear map from 512 dimensions to an adapter width w_{ad}, GELU, dropout, and a linear map back to 512 dimensions. Both output projections are zero-initialized, and both residual scales are initialized to one. The adapter is therefore exactly the identity map at initialization. Known-K runs use $w_{\mathrm{ad}} = 512$ and dropout 0.10, giving 1,052,674 trainable adapter parameters. Unknown-K runs use $w_{\mathrm{ad}} = 384$ for ImageNet-50/100 and $w_{\mathrm{ad}} = 768$ for ImageNet-200/1K, giving 790,274 and 1,577,474 parameters, respectively. Their dropout values are 0.15, 0.10, 0.12, and 0.12 in dataset order.

Because OpenCLIP remains frozen, we precompute eight stochastic feature views per image with FP16 autocast, store these views in FP16, and sample two distinct cached views at each neural update. Images are first bicubically resized to 256×256. For ImageNet-50/100/1K, each cached view then uses a bicubic random resized crop to 224×224 with scale $[0.45, 1]$ and aspect ratio $[0.75, 4/3]$, horizontal flipping with probability 0.5, color jitter $(0.3, 0.3, 0.2, 0.05)$ with probability 0.75, grayscale with probability 0.05, and Gaussian blur with probability 0.10. ImageNet-200 uses crop scale $[0.55, 1]$, aspect ratio $[0.8, 1.25]$, color jitter $(0.35, 0.35, 0.22, 0.06)$ with probability 0.80, grayscale with probability 0.03, and Gaussian blur with probability 0.05. OpenCLIP input normalization is applied last. The deterministic evaluation view is encoded and stored in FP32. It uses bicubic aspect-preserving resizing to 256 pixels, a 224×224 center crop, and the same OpenCLIP normalization. This fixed view is used for every full-dataset DP refresh and final cluster assignment.

The score model is trained on Gaussian perturbations of the normalized adapted features. Its density model therefore refers to the Gaussian-smoothed marginal p_σ for $\sigma > 0$ and does not require the unperturbed normalized features to admit a full-dimensional Euclidean density.

Neural optimization. The adapter, prototype, and score network are optimized with AdamW using $(\beta_1, \beta_2) = (0.9, 0.999)$. The learning rates in Table 7 are the initial effective parameter-group rates; the prototype group uses the same initial rate as the score network. All parameter groups follow a cosine schedule over the complete training horizon without optimizer warm-up. Generic gradient clipping is disabled.

Table 7: Dataset-specific optimization settings used in the main experiments. Feature batch denotes the number of cached feature vectors in each neural update. The adapter and score-network learning rates are reported separately.

(a) Known-K setting

Dataset	Feature batch	Epochs	Adapter LR	Score LR	Weight decay
ImageNet-50	192	400	1.0×10^{-5}	5.0×10^{-5}	1.0×10^{-5}
ImageNet-100	256	450	2.5×10^{-6}	5.0×10^{-5}	1.0×10^{-5}
ImageNet-200	512	240	2.5×10^{-6}	5.0×10^{-5}	1.0×10^{-5}
ImageNet-1K	512	180	2.5×10^{-6}	5.0×10^{-5}	1.0×10^{-5}

(b) Unknown-K setting

Dataset	Feature batch	Epochs	Adapter LR	Score LR	Weight decay
ImageNet-50	192	400	1.8×10^{-4}	1.8×10^{-4}	1.0×10^{-5}
ImageNet-100	256	450	2.5×10^{-4}	2.5×10^{-4}	1.0×10^{-6}
ImageNet-200	512	240	2.5×10^{-4}	2.5×10^{-4}	1.0×10^{-5}
ImageNet-1K	512	180	1.6×10^{-4}	1.6×10^{-4}	1.0×10^{-5}

The ImageNet-1K cache time is the measured sum of the eight-view training cache (6,733.6 seconds) and deterministic evaluation cache (3,656.8 seconds). The unknown-K ImageNet-1K run was resumed exactly from its fixed `last.pt` checkpoint after an external interruption. Its reported training time sums the two active process intervals, excludes downtime, and is rounded to the nearest minute. All known-K training times are measured directly using the complete dataset-specific horizons reported in Table 7; no timing is extrapolated from a shorter run.

As specified in Algorithm 1, $\mathcal{L}_{\text{score}}$ updates only the score network, with its feature targets detached from the adapter. The remaining encoder-dependent objectives update the adapter while OpenCLIP remains fixed. Known-K and unknown-K runs use dataset-wise training horizons of 400, 450, 240, and 180 epochs on ImageNet-50, ImageNet-100, ImageNet-200, and ImageNet-1K, respectively. Each horizon is fixed across all variants and seeds within the corresponding benchmark. Dataset-specific settings are summarized in Table 7. Batch size denotes the number of cached feature vectors used in one neural update rather than the image batch used during OpenCLIP feature extraction.

Score model and density-guided sampling. The noise-conditional score model first encodes $\log\sigma$ with a two-layer, 64-dimensional SiLU MLP. The resulting embedding is concatenated with the noisy 512-dimensional feature and passed through a two-hidden-layer SiLU MLP of width 256 without normalization. The final linear layer is zero-initialized. The score network contains 349,376 trainable parameters, and its raw output is converted to the score as $s_\psi(\widetilde{h},\sigma) = \mathrm{raw}_\psi(\widetilde{h},\log\sigma)/\sigma$.

During score training, σ is sampled log-uniformly from $[0.01, 0.15]$, with $\lambda(\sigma) = \sigma^2$ in equation B.1.6. Guidance uses three geometrically spaced noise levels from 0.15 to 0.01 and one score-guided update per level. At level σ_ℓ, the step size is $\epsilon_\ell = 0.01(\sigma_\ell/0.15)^2$. The stochastic exploration term in equation B.1.14 is $\gamma\sqrt{2\epsilon_\ell}z$, with $\gamma = 0.005$ for known-K and $\gamma = 0.003$ for unknown-K.

Because γ is smaller than the unit diffusion coefficient of standard Langevin sampling, this update is used as damped stochastic score-guided ascent rather than as an exact finite-step sampler from p_σ. The density-ascent analysis concerns its deterministic score drift, whose conditional mean is unaffected by the zero-mean exploration term. Each endpoint is ℓ_2-normalized before entering a cosine-based objective.

Semantic expansion and contrastive learning. Semantic expansion uses assignment temperature $\tau_f = 0.10$, at most $P = 8$ candidates per observed category, and confidence threshold $\delta = 0.30$ for ImageNet-50 known-K and 0.55 otherwise. Anchors and expanded semantic sets are refreshed once per epoch using the deterministic evaluation view. Each active FCL step samples one complete semantic category group containing at most 13 anchors, formed by five supports and eight candidates. All nonduplicate examples in the ordinary feature mini-batch remain in the denominator as unlabeled negatives. The FCL and PCL temperatures are $\tau = 0.10$ and $\tau_p = 0.10$, respectively. Each loss is averaged over its valid mini-batch elements. When no valid FCL positive pair is available, the implementation contributes exactly zero rather than constructing an artificial positive pair.

Variational DP inference. For known-K, we set $K_{\max} = K$ and retain all components. In the unknown-K protocol, the realized number of active clusters is neither supplied to nor fixed by the inference procedure. The finite truncation $K_{\max}$ specifies only the maximum number of available components. To provide the same relative capacity margin across the four ImageNet benchmark scales, we set $K_{\max} = 2K_{\text{ref}}$, where K_{ref} denotes the category count used for evaluation. This gives $K_{\max} = 100, 200, 400$, and 2000 for ImageNet-50, ImageNet-100, ImageNet-200, and ImageNet-1K, respectively. Accordingly, throughout this paper, "withheld" and "without a predefined K" mean that K_{ref} is not imposed as an optimization target, assignment constraint, active-set size, stopping criterion, or realized output cardinality; the posterior may return any active-set size permitted by the configured truncation. These terms do not mean that the finite-capacity configuration is independent of K_{ref}. Components are activated or removed solely from their posterior mixing weights, and the reported estimate is $\widehat{K} = |\mathcal{A}|$. Thus, the protocol specifies a loose capacity ceiling but does not specify the realized cluster count. All reported estimates lie below this ceiling, and its effect is evaluated in Appendix H.3.

The default active-component threshold is scaled with the capacity as $\varepsilon = 1/K_{\max}$, giving 0.01, 0.005, 0.0025, and 0.0005 in dataset order. The batch-prototype mass threshold is $\varepsilon_{\mathcal{B}} = 1$. The concentration parameter is $\alpha = 1$. The diagonal Normal-Gamma prior uses $m_0 = \mathbf{0}$, $\beta_0 = 10^{-3}$, $a_0 = 1$, and $b_{0d} = 1/512$ for every feature dimension d. The mixture operates on the ambient coordinates of the normalized features to compute posterior responsibilities and local guidance directions; it is not used to model the original image distribution.

In the known-K setting, every posterior refresh initializes $q(Z)$ from the current feature geometry by deterministic spherical k-means++ on a seed-fixed sample of at most 20,000 normalized features, followed by ten spherical k-means iterations and hard one-hot assignment of all examples. This per-refresh initialization keeps every supplied component recoverable as the learned representation changes.

We use the same deterministic per-refresh rejuvenation in the unknown-K setting. It gives every truncated component a birth proposal based only on the current feature geometry, after which the posterior mixing weights may retain or prune it. This prevents a zero-mass component in an earlier mean-field solution from becoming permanently unrecoverable while preserving posterior-based component activation and removal.

One outer training iteration in Algorithm 1 corresponds to one epoch. At the beginning of each epoch, the complete deterministic feature collection is recomputed and the variational posterior is refreshed. CAVI performs three fixed sweeps per refresh without an ELBO-based early-stopping rule. Responsibilities and sufficient statistics are processed in blocks of 4,096 samples.

For final label-guided known-K inference, we apply support-anchored spherical k-means to the normalized features of the complete training pool. The observed-category centroids are initialized from normalized support means, and the remaining centroids are selected by anchor-aware k-means++. During each of 30 Lloyd iterations, assignments are made by cosine similarity, every support retains its supplied category assignment, and each support enters the corresponding centroid update with effective weight 100. Centroids are ℓ_2-normalized after each update. We run three restarts with seeds offset from the run seed and select the restart maximizing the mean assignment similarity plus 0.05 times support-anchor accuracy. Support samples are excluded from metric computation. DD-FSC-U and the known-K *w/o FCL* configuration instead run standard Euclidean k-means on the ℓ_2-normalized features, with 20 k-means++ initializations and at most 300 Lloyd iterations. They use no support labels, anchors, or label-derived constraints during training or final inference. Because their centroids are not renormalized after each update, we do not describe this procedure as spherical k-means. Ground-truth evaluation labels are accessed only after the assignments have been fixed and are used exclusively to compute the metrics. For final unknown-K inference, we refresh the variational posterior for ten CAVI sweeps and predict from its active components.

DP–diffusion coupling uses $\lambda_{\text{DP}} = 0.2$. Before coupling, the responsibility-weighted DP field in equation 5 is converted into the bounded normalized direction used in equation 6, with $\varepsilon_{\text{norm}} = 10^{-12}$. This separates λ_{DP} from the absolute Normal-Gamma precision scale while preserving the responsibility- and covariance-dependent direction. For joint neural optimization, the ELBO is normalized as $\bar{\mathcal{L}}_{\text{ELBO}} = \mathcal{L}_{\text{ELBO}}/(Nd_h)$. The loss coefficients in equation 8 are $(\lambda_{\text{score}}, \lambda_{\text{ELBO}}, \lambda_{\text{FCL}}, \lambda_{\text{DGCL}}, \lambda_{\text{PCL}}) = (0.05, 0.5, 0.02, 0.01, 0.005)$. The normalization removes

the dependence of the ELBO scale on the dataset size and feature dimension. Because equation 8 minimizes the negative normalized ELBO, λ_{ELBO} is nonnegative.

C.3 Computational Complexity and Resource Footprint

DD-FSC couples score-guided representation learning with variational DP inference, making it important to distinguish the total representation-learning cost from the incremental cost of adaptive cluster discovery. We therefore analyze its asymptotic time and working-memory complexity and complement this analysis with wall-clock, peak-memory, and component-wise measurements. Frozen-feature caching avoids repeated backbone evaluation, while chunked variational updates avoid materializing a dense $N \times K_{\max}$ responsibility matrix. These design choices keep the trainable-stage memory footprint compact and limit the additional cost of unknown-K inference without treating the method-specific score computation as negligible.

Asymptotic time and memory complexity. Let N denote the number of training samples, d_h the feature dimension, B the neural mini-batch size, and $K_{\max}$ the DP truncation level. Let L be the number of noise levels, S_{CAVI} the number of CAVI sweeps, T_{km} the number of spherical k-means iterations, $N_s = \min(N, 20000)$ the initialization-sample size, and $T_{\mathrm{L}} = \sum_{\ell=1}^{L} M_\ell$ the total number of score-guided updates per trajectory. In our implementation, $S_{\mathrm{CAVI}} = 3$, $T_{\mathrm{km}} = 10$, and $T_{\mathrm{L}} = 3$ during training.

For the diagonal Normal–Gamma mixture, one CAVI sweep evaluates the component likelihoods and sufficient statistics in $\mathcal{O}(NK_{\max}d_h)$ time. The deterministic posterior rejuvenation additionally requires $\mathcal{O}(T_{\mathrm{km}}N_sK_{\max}d_h)$ operations for spherical k-means initialization and $\mathcal{O}(NK_{\max}d_h)$ operations to assign the complete feature collection. The resulting cost of one posterior refresh is therefore

$$\mathcal{O}((S_{\mathrm{CAVI}} + 1)NK_{\max}d_h + T_{\mathrm{km}}N_sK_{\max}d_h)\,. \tag{C.2}$$

Thus, for fixed $K_{\max}$ and d_h, the posterior refresh scales linearly with the number of samples. Its dependence on the truncation capacity remains explicit and linear in $K_{\max}$.

For neural optimization, let C_{ad} and C_{score} denote the cost of one adapter and score-network evaluation, respectively, and let $K_{\mathcal{B}} = |\mathcal{A}_{\mathcal{B}}|$. A mini-batch update has the upper-bound complexity

$$\mathcal{O}\big(BC_{\mathrm{ad}} + \big[(T_{\mathrm{L}} + 1)B + T_{\mathrm{L}}K_{\mathcal{B}}\big]C_{\mathrm{score}} + \big(B^2 + BK_{\mathcal{B}} + K_{\mathcal{B}}^2\big)d_h\big)\,. \tag{C.3}$$

The three grouped terms correspond to adapter evaluation; score-network calls for score matching and sample or prototype trajectories; and feature-space operations for mini-batch contrast, posterior-weighted prototype construction, and prototype contrast, respectively. Let C_{batch} denote the mini-batch cost in equation C.3. Combining the neural updates with the full-dataset posterior refresh gives the per-epoch complexity

$$\mathcal{O}\left(\left\lceil \frac{N}{B} \right\rceil C_{\mathrm{batch}} + (S_{\mathrm{CAVI}} + 1)NK_{\max}d_h + T_{\mathrm{km}}N_sK_{\max}d_h\right), \tag{C.4}$$

For final label-guided known-K inference, let $R = 3$ be the number of restarts and $T_f = 30$ the number of iterations. Support-anchored spherical k-means costs $\mathcal{O}(RT_fNKd_h)$. DD-FSC-U uses $R_u = 20$ standard k-means initializations and at most $T_u = 300$ iterations, giving $\mathcal{O}(R_uT_uNKd_h)$ in the worst case. The ten-sweep variational refresh under unknown-K costs $\mathcal{O}(10NK_{\max}d_h)$.

Responsibilities are processed in blocks of $B_{\mathrm{DP}} = 4096$, so a dense $N \times K_{\max}$ responsibility matrix need not be materialized. Excluding the stored feature views, the principal working-memory cost is

$$\mathcal{O}\big(Nd_h + B_{\mathrm{DP}}K_{\max} + K_{\max}d_h + B^2 + K_{\mathcal{B}}d_h\big)\,. \tag{C.5}$$

The DP posterior introduces no additional trainable neural parameters. Including the score network, DD-FSC has 1,402,050 trainable parameters under known-K, 1,139,650 under unknown-K on ImageNet-50/100, and 1,926,850 under unknown-K on ImageNet-200/1K. The raw feature-cache payload is $20Nd_h$ bytes because each image stores eight FP16 training views and one FP32 evaluation view. With $d_h = 512$, this corresponds to approximately 0.620, 1.240, 2.480, and 12.218 GiB for ImageNet-50, ImageNet-100, ImageNet-200, and ImageNet-1K, respectively.

Table 8: Wall-clock computational cost of one complete DD-FSC run on a single 12-GiB NVIDIA GeForce RTX 3080 Ti GPU.

(a) Known-K setting

Dataset	Caching time	Training time	Inference time	Peak memory
ImageNet-50	22 min 37.5 s	51 min 21.1 s	8.2 s	118.2 MiB
ImageNet-100	35 min 7.5 s	93 min 12.5 s	16.5 s	118.6 MiB
ImageNet-200	27 min 13.8 s	32 min 9.2 s	16.7 s	119.5 MiB
ImageNet-1K	2 h 53 min	4 h 24 min	2 min 3.6 s	129.8 MiB

(b) Unknown-K setting

Dataset	Caching time	Training time	Inference time	Peak memory
ImageNet-50	22 min 37.5 s	54 min 36.6 s	10.9 s	114.5 MiB
ImageNet-100	35 min 7.5 s	95 min 19.8 s	18.5 s	115.2 MiB
ImageNet-200	27 min 13.8 s	31 min 22.8 s	23.6 s	134.3 MiB
ImageNet-1K	2 h 53 min	4 h 47 min	3 min 14.2 s	194.7 MiB

End-to-end resource profile. Each computational profile in Table 8 is measured from one complete run and is not averaged across seeds. The unknown-K profiles use seed 42. Every run uses a single 12-GiB NVIDIA GeForce RTX 3080 Ti GPU. Experiments use PyTorch 2.7.1+cu118, CUDA 11.8, and cuDNN 9.1.0. The eight stochastic training views use FP16 autocast and FP16 storage. The deterministic center-crop cache and all trainable DD-FSC paths use FP32.

DD-FSC performs all trainable optimization in the cached feature space. The frozen OpenCLIP encoder is therefore evaluated only during one-time cache construction rather than once per training epoch. This design converts repeated visual encoding into a one-time preprocessing cost and keeps the subsequent score learning and DP inference entirely within the feature space. We accordingly measure caching and training times separately. Caching time sums feature extraction for the eight stochastic training views and the deterministic center-crop view. Training time excludes feature caching but includes the epoch-wise DP posterior updates. Inference time includes the setting-specific final clustering procedure and cluster assignment. Peak memory is the maximum GPU memory allocated during the trainable feature-space stage and excludes the separate OpenCLIP feature-extraction stage. Table 8 reports the resulting computational profile.

Table 8 shows that the feature-space design remains practical as the dataset scale increases. On ImageNet-1K, the post-caching training time is 4 h 24 min under known-K and 4 h 47 min under unknown-K, while final inference requires only 2 min 3.6 s and 3 min 14.2 s, respectively. The trainable feature-space stage uses at most 194.7 MiB of GPU memory across all reported configurations. Moreover, in the measured runs, unknown-K training is at most 8.7% slower than its known-K counterpart across the four datasets, despite jointly inferring the active cluster structure. These results demonstrate a compact trainable-stage memory footprint and a controlled runtime increase for adaptive cluster inference on a single commodity GPU.

Component-wise runtime. To identify the main sources of computational cost, we profile DD-FSC on ImageNet-100 using a single 12-GiB NVIDIA GeForce RTX 3080 Ti GPU. Each time is reported as the mean $\pm$ sample standard deviation over ten fully active epochs from one prespecified run. The first seven warm-up and loss-ramp epochs are excluded. Feature caching is also excluded because it is performed only once and shared by all training stages. The reported shares are computed relative to the synchronized per-epoch training time in the corresponding setting. Shared backward and optimizer operations are assigned to the core optimization path. Because synchronization barriers introduce diagnostic timing overhead, these per-epoch measurements are used only for component-wise comparison and are not extrapolated to replace the end-to-end measurements in Table 8.

Table 9 separates the computation associated with the two principal mechanisms of DD-FSC. Score matching and guided trajectories account for 39.7% and 42.4% of the synchronized profiled-epoch time under known-K and unknown-K, respectively. In contrast, DP initialization and CAVI refresh account for only 6.2% and 7.8%. Thus, although unknown-K introduces nonparametric posterior inference, the DP path remains a small fraction of the total runtime. This is an important practical

Table 9: Component-wise runtime profile of DD-FSC on ImageNet-100 using a single 12-GiB NVIDIA GeForce RTX 3080 Ti GPU.

Component	Known-K		Unknown-K	
	Time/epoch	Share	Time/epoch	Share
Core adapter and loss optimization	5.37 ± 0.17 s	40.7%	5.43 ± 0.22 s	37.3%
Score matching and guided trajectories	5.23 ± 0.13 s	39.7%	6.17 ± 0.18 s	42.4%
DP initialization and CAVI refresh	0.82 ± 0.10 s	6.2%	1.14 ± 0.16 s	7.8%
Data movement, checkpointing, and other operations	1.75 ± 0.07 s	13.3%	1.82 ± 0.16 s	12.5%
Synchronized profiled epoch	$\mathbf{13.18\pm0.27}$ s	**100%**	$\mathbf{14.57\pm0.49}$ s	**100%**

consequence of the truncated and chunked variational design: DD-FSC obtains adaptive cluster-number inference without making posterior refresh the dominant computational bottleneck.

Although the unknown-K configuration doubles the truncation capacity from $K_{\max} = 100$ to $K_{\max} = 200$, its synchronized profiled epoch is only 10.6% slower than that of the known-K configuration. Together with the end-to-end measurements in Table 8, this result reveals a targeted cost–performance trade-off. DD-FSC concentrates its method-specific computation on score-guided representation refinement, while the chunked posterior updates keep adaptive partition inference below 8% of the profiled-epoch time. The method therefore does not eliminate the cost of jointly refining representations and partitions; instead, it limits the incremental cost of unknown-K inference and avoids repeated backbone computation. This combination enables the complete known- and unknown-K pipelines to operate on a single 12-GiB NVIDIA GeForce RTX 3080 GPU.

D Additional Results on Image Clustering Benchmarks

We extend the evaluation beyond the ImageNet benchmarks in the main paper to six datasets spanning handwritten digits, fashion products, generic objects, and fine-grained dog breeds. Their distinct scales and category granularities test whether DD-FSC transfers across feature distributions rather than relying on ImageNet-specific structure. We use the same protocols, metrics, and baseline-sourcing policy as Section 4.1; DD-FSC-U denotes the fully unsupervised variant.

Generalization under known K. Table 10 compares the two supervision regimes on four additional benchmarks. DD-FSC-U ranks first on six of the 12 metrics and among the top two on nine, including all three metrics on ImageNet-Dogs. With partial-category five-shot supervision, DD-FSC ranks first on 10 metrics and achieves the best ACC on every benchmark. The improvement persists from coarse-grained CIFAR categories to fine-grained dog breeds. This pattern is consistent with complementary roles for the two signals: density-guided refinement provides a strong unsupervised partition, while sparse supports improve that partition without requiring labels for every category. The results therefore extend the evidence for density-guided semantic propagation beyond the ImageNet settings studied in the main paper.

Adaptive clustering under unknown K. Table 11 evaluates assignment quality when the partition cardinality must also be inferred. DD-FSC ranks first or second in NMI on all four benchmarks and in ACC and ARI on three. It achieves the best result for all three metrics on Fashion-MNIST and the best ACC and ARI on CIFAR-20. These results are obtained without providing the ground-truth cardinality. The consistent rankings across digits, fashion images, and natural images show that adaptive partition recovery remains effective when the active components and feature representation must be learned jointly.

Calibration of the inferred partition. Table 12 examines whether the assignment results are accompanied by a correctly scaled partition. DD-FSC gives the closest mean estimate on CIFAR-10 and CIFAR-20, the second-closest on Fashion-MNIST, and ties DeepDP-TGMM for the second-closest estimate on MNIST. Every mean estimate is within 1.7 clusters of the ground-truth count. Read together with Table 11, these estimates show that the clustering scores do not arise from a severely over- or under-segmented solution. They further indicate that DP component selection remains calibrated after transfer from ImageNet to smaller benchmarks, which is central to using DD-FSC as a genuinely nonparametric clustering model.

Table 10: Known-K clustering results on standard-scale image benchmarks. DD-FSC-U denotes the fully unsupervised variant of DD-FSC. Best results are in **bold** and second best are underlined.

Methods	CIFAR-10			CIFAR-20			STL-10			ImageNet-Dogs		
	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI
(a) Unsupervised clustering												
SIC (AAAI, 2023)	84.8	92.7	84.6	59.3	58.4	44.0	95.4	98.1	95.9	71.8	71.3	58.6
TAC (ICML, 2024)	83.3	91.9	83.1	61.1	60.7	44.8	95.5	98.3	96.1	80.6	83.0	72.2
CPP (ICLR, 2024)	84.6	85.5	82.8	69.3	66.4	42.4	95.6	95.5	90.3	82.1	73.5	64.7
CLUDI (ICML, 2025)	**89.6**	**95.3**	89.8	59.5	67.3	**57.1**	96.8	98.7	97.1	84.3	85.2	78.3
SAC (AAAI, 2026)	88.5	94.9	89.1	66.1	67.4	51.9	96.2	98.5	96.7	83.7	86.5	75.9
GCPM-G (CVPR, 2026)	87.5	94.1	84.9	68.4	68.7	47.1	96.9	98.8	97.4	85.1	88.1	74.1
MiniClustering (ICLR, 2026)	84.5	91.1	82.5	69.9	68.8	45.5	95.6	97.9	97.6	87.9	90.1	77.1
CDC+SANE (ICLR, 2026)	88.7	94.7	89.0	68.5	62.7	46.6	86.9	93.6	86.8	78.2	84.3	73.8
SATC (ICLR, 2026)	88.9	94.5	88.3	70.5	63.4	48.1	**97.3**	**99.0**	**97.9**	89.7	91.4	86.7
DD-FSC-U (Ours)	89.4	95.1	**90.1**	**71.2**	**70.8**	52.6	96.8	98.4	96.9	**90.2**	**92.8**	**88.6**
(b) Few-shot clustering (30% of categories, 5 shots each)												
Soft k-Means (ICLR, 2018)	84.1	90.6	81.8	53.7	59.5	36.9	86.9	93.0	86.0	86.2	80.5	72.5
cluster-FSL (CVPR, 2022)	85.2	90.8	82.2	64.4	70.8	51.5	91.2	96.9	92.1	88.2	83.4	77.1
UNEM (CVPR, 2025)	86.0	92.3	91.8	66.7	70.3	52.4	94.3	98.1	95.2	88.0	86.1	83.9
EfficientFSL (AAAI, 2026)	91.9	96.2	92.6	70.1	69.6	51.6	95.5	98.5	96.1	91.3	90.7	82.1
TIM++ (AAAI, 2026)	91.8	96.5	91.2	69.6	70.7	**57.2**	96.4	98.3	96.8	92.1	92.3	85.2
OSESSL (ICLR, 2026)	**92.3**	96.3	92.3	76.5	73.4	56.6	94.7	97.8	96.2	93.2	93.9	87.1
DD-FSC (Ours)	91.6	**97.1**	**92.9**	**79.8**	**73.8**	56.4	**97.2**	**98.7**	**97.2**	**93.6**	**94.8**	**89.1**

Table 11: Unknown-K clustering results on standard-scale image benchmarks. Best results are in **bold** and second best are underlined.

Methods	MNIST			Fashion-MNIST			CIFAR-10			CIFAR-20		
	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI
DeepDPM (CVPR, 2022)	94.0	**98.0**	95.0	68.0	62.0	51.0	63.9	67.7	66.8	43.5	37.6	21.6
UNSEEN+DCN (ICDMW, 2024)	87.8	86.0	83.7	63.7	57.3	46.9	68.1	70.7	64.6	39.6	40.5	23.6
SpherePair (NeurIPS, 2025)	91.2	96.6	92.6	71.4	73.7	60.2	71.7	89.8	71.0	46.5	37.0	26.1
SCMax (AAAI, 2026)	89.9	87.5	87.1	70.7	72.8	66.4	75.4	83.3	72.4	47.2	45.6	28.5
DeepDP-TGMM (CVPR, 2026)	94.1	97.8	**95.3**	72.1	77.3	68.5	80.0	84.0	**75.0**	**49.0**	42.0	25.0
DD-FSC (Ours)	**95.0**	97.0	94.9	**74.6**	**77.8**	**70.3**	**82.7**	**90.7**	74.2	48.3	**48.5**	**31.4**

Table 12: Estimated cluster count $\widehat{K}$ on standard-scale image benchmarks under the unknown-K setting. Best results are in **bold** and second best are underlined.

Methods	MNIST	Fashion-MNIST	CIFAR-10	CIFAR-20
DeepDPM (CVPR, 2022)	$\mathbf{10.0 \pm 0.0}$	$\mathbf{10.2 \pm 0.8}$	14.7 ± 1.7	14.0 ± 1.4
UNSEEN+DCN (ICDMW, 2024)	12.3 ± 0.9	13.3 ± 1.2	17.3 ± 1.2	29.3 ± 3.4
SpherePair (NeurIPS, 2025)	11.7 ± 0.5	11.7 ± 0.9	12.0 ± 0.8	24.0 ± 0.8
SCMax (AAAI, 2026)	8.7 ± 0.9	8.3 ± 1.2	8.3 ± 0.9	17.7 ± 0.9
DeepDP-TGMM (CVPR, 2026)	10.3 ± 0.5	9.0 ± 0.8	11.4 ± 0.6	18.0 ± 0.7
DD-FSC (Ours)	9.7 ± 0.5	10.3 ± 0.5	$\mathbf{10.3 \pm 0.5}$	$\mathbf{18.3 \pm 1.2}$

E Ablation Protocols and Cross-Benchmark Analysis

This section specifies the implementation of each variant in Table 4 and extends the same study to the remaining ImageNet benchmarks.

Each variant is initialized and trained separately. Within a seed, all variants use the same train and evaluation partitions, observed categories, and exact five-shot support samples. The backbone initialization, augmentations, optimizer, number of epochs, update frequency, and total optimization budget are also fixed. We reuse the full-model hyperparameters without variant-specific retuning. Thus, within each paired run, the stated intervention is the only intended difference between a variant and its reference configuration.

Known-K experiments use the reference cluster count only where the protocol supplies K. In unknown-K experiments, the reference K is not used for training, hyperparameter or threshold selection, checkpoint selection, DP truncation, initialization, or active-component decisions. Ground-

truth identities of unlabeled samples are used only after inference to compute the evaluation metrics. All variants use the same final-checkpoint rule, evaluation samples, and Hungarian matching for ACC. NMI, ACC, and ARI are averaged over three paired runs. For unknown-K, each seed produces an integer-valued $\widehat{K} = |\mathcal{A}|$; we report its mean and population standard deviation over the same three runs.

E.1 Unknown-K Ablation Variants

Guidance controls. These controls retain $\mathcal{L}_{\mathrm{FCL}}$ and vary the construction of its semantic sets. To isolate this factor, $\mathcal{L}_{\mathrm{DGCL}}$ and $\mathcal{L}_{\mathrm{PCL}}$ are disabled throughout this block. *Support only* sets $\mathcal{C}_y = \varnothing$ and forms semantic positives exclusively from the labeled supports. *kNN expansion* replaces score-guided retrieval with nearest-neighbor retrieval based on cosine similarity in the normalized current feature space. It uses the same per-category candidate budget and refresh frequency as the score-based controls.

Noise only does not train the score network and sets $s_\psi(\cdot, \sigma) = 0$ in equation 1, but preserves the initialization, noise levels, step sizes, trajectory length, stochastic terms, and top-P cap. *Score only* retains learned score trajectories, category anchors, predicted-category agreement, and the top-P rule but sets $\delta = 0$, selecting up to P highest-ranked samples assigned to each observed category without confidence rejection. *Score + filtering* instead uses fixed $\delta > 0$. The variants share only the upper bound P: if fewer than P candidates pass, filtering leaves slots empty rather than adding lower-confidence assignments. Thus, their difference measures confidence gating's full reliability–coverage trade-off, not candidate quality at fixed cardinality. The values of δ and P are fixed before evaluation; no unlabeled ground-truth label or clustering metric is used for selection.

The ablation reports the downstream clustering metrics produced by each selection rule. Unlabeled ground-truth identities are not used to construct, tune, or filter the candidate sets.

Objective and supervision ablations. These variants start from the complete DD-FSC configuration. *w/o FCL*, *w/o DGCL*, and *w/o PCL* set λ_{FCL}, λ_{DGCL}, and λ_{PCL} to zero, respectively, while leaving the remaining objectives, update frequencies, and budgets unchanged. For *w/o FCL*, FCL is the only training objective that consumes the labeled and expanded semantic sets. We therefore disable semantic expansion and pass no support identity, anchor, or label-derived constraint to training or final inference. Under known-K, this configuration is exactly DD-FSC-U and uses the same standard Euclidean k-means predictor with 20 k-means++ initializations and at most 300 Lloyd iterations. Consequently, the DD-FSC-U entries in Table 1 and the known-K *w/o FCL* entries in Tables 4 and 13–15 report the same runs rather than independent experiments. This control measures the complete contribution of the label-guided path, including FCL training and support-anchored final inference; it does not isolate FCL training under a fixed predictor.

With *w/o DGCL*, score learning and score-guided semantic expansion remain active, and the support-anchored predictor is retained. With *w/o PCL*, the supplied known-K is still used to maintain the prototype set, but prototype contrast does not update the encoder; the support-anchored predictor is again retained. These two controls therefore isolate their respective representation objectives. *Full DD-FSC* retains all objectives and the complete semantic-expansion procedure.

E.2 Unknown-K Ablation Variants

Representation ablations. *w/o score guidance* removes $\mathcal{L}_{\mathrm{score}}$, does not train the score network, and sets $s_\psi(\cdot, \sigma) = 0$ in every score-dependent trajectory. This change is applied consistently to semantic expansion, DGCL counterpart construction, sample-level diffusion, and prototype-level diffusion. The same stochastic terms, number of steps, step sizes, and update budgets are retained. DP guidance remains available in the coupled unknown-K trajectory. This is therefore a system-level removal rather than a single loss-weight change.

w/o semantic expansion sets $\mathcal{C}_y = \varnothing$ for all observed categories. FCL remains active on the original supports, while score learning, DGCL, DP inference, and PCL are unchanged. *w/o FCL* sets $\lambda_{\mathrm{FCL}} = 0$ and does not use labeled or expanded semantic sets in any other objective. *w/o DGCL* sets $\lambda_{\mathrm{DGCL}} = 0$ but retains score learning, score-guided semantic expansion, FCL, and all DP-dependent components.

Table 13: Ablation study of DD-FSC on ImageNet-50.

(a) Known-K setting

Variant	NMI	ACC	ARI
Semantic-expansion controls			
Support only	81.3	71.2	64.3
kNN expansion	83.0	73.3	68.9
Noise only	84.5	74.6	68.8
Score only	87.3	82.6	75.7
Score + filtering	91.8	85.9	79.4
Training-objective ablations			
w/o FCL	90.8	88.2	85.6
w/o DGCL	92.2	82.9	79.2
w/o PCL	92.7	86.7	82.2
Full DD-FSC	95.5	93.4	87.3

(b) Unknown-K setting

Variant	NMI	ACC	ARI	$\widehat{K}$
Representation-learning ablations				
w/o score guidance	72.2	64.2	53.6	37.3±3.3
w/o semantic expansion	72.9	66.1	54.4	65.3±0.9
w/o FCL	73.3	66.9	57.3	62.7±1.2
w/o DGCL	71.8	71.1	58.1	58.0±6.4
Adaptive-clustering ablations				
w/o joint DP learning	66.2	64.9	55.2	39.7±1.7
w/o DP–diffusion coupling	71.2	67.0	57.1	56.7±8.3
w/o adaptive updates	73.1	69.5	59.1	61.0±0.8
w/o PCL	77.8	71.3	60.4	58.7±11.9
Full DD-FSC	82.9	76.0	66.8	51.7±0.9

Inference ablations. *w/o joint DP learning* is a two-stage control. The encoder and score network are first trained with FCL and DGCL, but without DP guidance or PCL. The representation is then frozen, and the same variational DP model is fitted to the frozen features. Consequently, posterior responsibilities, component parameters, and the active set cannot feed back into representation learning.

w/o DP–diffusion coupling sets $\lambda_{\mathrm{DP}} = 0$ in equation 6 for both sample and prototype trajectories. Score diffusion, variational DP updates, active-component updates, and the remaining objectives stay active. PCL therefore uses score-guided prototype counterparts without the DP guidance field. *w/o adaptive updates* freezes the active component set after the warm-up stage. Components can neither enter nor leave this frozen set, but posterior responsibilities and component parameters over the frozen components remain trainable. The warm-up set is obtained with the same posterior-mass rule as the full model, and neither its construction nor the truncation level uses the reference K. *w/o PCL* sets $\lambda_{\mathrm{PCL}} = 0$ while retaining semantic expansion, $\mathcal{L}_{\mathrm{FCL}}$, $\mathcal{L}_{\mathrm{DGCL}}$, DP–diffusion coupling, and adaptive component updates. Sample-level coupling remains active: equation 6 produces detached guided counterparts for ordinary mini-batch samples, which enter $\mathcal{L}_{\mathrm{DGCL}}$, even though no guided prototype target is consumed by $\mathcal{L}_{\mathrm{PCL}}$. The DP posterior still determines $\widehat{K}$ in all applicable variants. *Full DD-FSC* retains all representation and adaptive-inference components.

E.3 Additional benchmark ablations

Tables 13– 15 test whether the ImageNet-100 findings in Table 4 persist as the category space grows from 50 to 1,000 classes. The paired interventions, five-shot splits, training budgets, and evaluation rules remain unchanged. The analysis therefore asks whether semantic propagation and complementary objectives remain effective across scale, and whether unknown-K variants recover both cluster counts and sample assignments.

Scale-consistent semantic propagation. Across all four benchmarks, learned score guidance improves NMI, ACC, and ARI over both kNN expansion and matched noise. These controls use the same candidate budget and refresh schedule, isolating the learned density direction from local neighbors and stochastic perturbations. Confidence filtering adds a second gain at every scale under the shared top-P cap. Relative to score guidance alone, it adds 3.2–4.6 NMI, 3.3–5.6 ACC, and 3.1–6.7 ARI points. Even on ImageNet-1K, where the NMI difference between the unfiltered guidance controls is small, the ACC and ARI gains remain clear. Since filtering may select fewer than P candidates, this persistent ordering supports rejecting unreliable assignments rather than filling unused slots; it measures a reliability–coverage trade-off, not fixed-cardinality quality.

Complementary learning signals. The known-K *w/o FCL* endpoint coincides with DD-FSC-U and therefore measures the complete removal of the label-guided path rather than FCL training alone. Restoring this path improves every metric across all four benchmark scales. Among the objective-only controls, removing DGCL produces the largest ACC and ARI drop at every

Table 14: Ablation study of DD-FSC on ImageNet-200.

(a) Known-K setting

Variant	NMI	ACC	ARI
Semantic-expansion controls			
Support only	74.4	67.2	53.3
kNN expansion	75.0	68.2	54.2
Noise only	75.2	70.7	54.5
Score only	80.1	75.6	60.1
Score + filtering	84.7	80.2	65.7
Training-objective ablations			
w/o FCL	87.8	81.9	67.7
w/o DGCL	84.4	75.9	64.0
w/o PCL	82.6	78.8	67.5
Full DD-FSC	88.7	83.7	71.4

(b) Unknown-K setting

Variant	NMI	ACC	ARI	$\widehat{K}$
Representation-learning ablations				
w/o score guidance	56.8	32.5	26.8	157.7±1.2
w/o semantic expansion	57.3	34.3	28.8	244.3±4.5
w/o FCL	57.2	37.5	28.3	212.7±3.3
w/o DGCL	59.9	42.4	27.5	216.3±1.7
Adaptive-clustering ablations				
w/o joint DP learning	54.1	35.9	23.1	174.7±1.9
w/o DP–diffusion coupling	57.4	39.5	27.1	221.0±1.4
w/o adaptive updates	56.3	40.0	27.7	245.0±4.9
w/o PCL	60.3	42.9	31.4	216.0±1.4
Full DD-FSC	63.6	46.7	35.3	198.7±1.9

Table 15: Ablation study of DD-FSC on ImageNet-1K.

(a) Known-K setting

Variant	NMI	ACC	ARI
Semantic-expansion controls			
Support only	77.8	57.6	35.6
kNN expansion	78.9	57.9	38.1
Noise only	79.0	58.1	37.4
Score only	79.2	61.3	40.5
Score + filtering	82.4	65.8	43.6
Training-objective ablations			
w/o FCL	82.4	70.3	48.3
w/o DGCL	81.6	67.4	45.4
w/o PCL	83.1	67.5	46.8
Full DD-FSC	86.4	72.4	51.2

(b) Unknown-K setting

Variant	NMI	ACC	ARI	$\widehat{K}$
Representation-learning ablations				
w/o score guidance	52.3	24.9	18.5	769.7±15.0
w/o semantic expansion	53.2	25.5	20.8	1295.3±11.6
w/o FCL	54.0	27.5	21.9	1148.0±20.8
w/o DGCL	56.1	30.7	23.2	1058.7±10.6
Adaptive-clustering ablations				
w/o joint DP learning	50.9	24.8	20.7	921.7±4.9
w/o DP–diffusion coupling	52.2	28.6	22.2	1165.3±12.0
w/o adaptive updates	55.3	29.4	22.1	1284.3±15.0
w/o PCL	57.5	32.2	24.9	1064.7±12.7
Full DD-FSC	60.5	36.6	28.1	1049.0±10.2

scale: 5.0–10.5 and 5.8–9.7 points, respectively. PCL exhibits a distinct metric signature through prototype-level separation. These results separate the contribution of sparse semantic supervision from the complementary roles of density-guided sample alignment and prototype structure.

Scale-normalized behavior under unknown K. Relative cluster-count error reveals a stable cross-scale pattern. Removing score guidance produces 21–25% under-segmentation, while the two-stage variant without joint DP learning also under-segments every benchmark. Removing semantic expansion or freezing the active set reverses the sign, producing approximately 20–31% and 22–28% over-segmentation, respectively. Removing the DP–diffusion field likewise causes 10–17% over-segmentation. These proportional errors persist over a twenty-fold change in K, ruling out a fixed additive component-count bias. They instead show that density guidance, semantic propagation, and posterior adaptation exert complementary control over partition granularity. The complete model keeps the relative cluster-count error below 5% on all four benchmarks.

Cluster count and assignment quality. The unknown-K results show that cluster-count accuracy alone does not guarantee a correct partition. On ImageNet-1K, removing DGCL yields 1058.7 clusters yet trails DD-FSC by 5.9 ACC and 4.9 ARI points. In every ablation table, only full DD-FSC obtains both the best clustering metrics and the closest count. Since budgets are shared and variants are not retuned, this joint advantage is not explained by variant-specific active-set threshold tuning. It instead supports the closed loop: density-aware features refine posterior assignments, which return component guidance to the encoder. Its persistence from 50 to 1,000 categories shows that this mechanism is not an ImageNet-100-specific correction.

Table 16: Run-to-run variability on ImageNet-100 and ImageNet-200. Results are reported as mean $\pm$ sample standard deviation over ten new runs. Higher values are better for NMI, ACC, and ARI, while lower values are better for $|\widehat{K} - K|$. The inferred $\widehat{K}$ is reported descriptively and is not directly included in the hypothesis tests.

(a) Known-K setting

Dataset	Method	NMI	ACC	ARI
ImageNet-100	TIM++	92.64±0.85	83.60±0.82	70.92±1.20
	OSESSL	91.37±0.84	84.06±0.83	72.69±1.06
	DD-FSC (Ours)	92.36±0.49	85.19±0.72	77.25±1.02
ImageNet-200	TIM++	87.45±0.88	78.15±0.87	69.51±1.18
	OSESSL	88.92±0.82	79.26±1.30	70.79±1.61
	DD-FSC (Ours)	88.64±0.57	83.53±0.81	71.33±0.87

(b) Unknown-K setting

Dataset	Method	NMI	ACC	ARI	$\widehat{K}$	$\lvert\widehat{K} - K\rvert$
ImageNet-100	DeepDP-TGMM	68.88±0.88	53.10±1.24	46.24±1.25	105.60±2.01	5.60±2.01
	DD-FSC (Ours)	69.63±0.64	55.45±0.95	44.69±0.81	103.40±2.01	3.40±2.01
ImageNet-200	DeepDP-TGMM	60.38±1.28	45.28±1.25	32.84±1.28	200.80±5.53	4.20±3.43
	DD-FSC (Ours)	62.91±0.85	46.69±0.82	34.82±1.53	199.10±2.77	2.30±1.64

F Run-to-Run Variability and Statistical Significance

The three-run averages in Tables 1–3 provide broad benchmark comparisons but do not fully characterize uncertainty across training runs. Randomness arises from observed-category and support selection, model initialization, mini-batch sampling, data augmentation, Langevin noise, and DP initialization. We therefore conduct an independent ten-run analysis to determine whether the observed differences exceed this run-to-run variation.

The analysis covers ImageNet-100 and ImageNet-200 under both known-K and unknown-K settings. Under known-K, DD-FSC is compared with TIM++ (Li et al., 2026a) and OSESSL (Cui et al., 2026). These methods jointly provide the strongest baseline results across NMI, ACC, and ARI on the selected benchmarks. Under unknown-K, DD-FSC is compared with DeepDP-TGMM (Tepakbong et al., 2026), the strongest competing method on these benchmarks that also infers the cluster count. The datasets, methods, metrics, and random seeds are fixed before running this analysis and are not changed according to its outcomes.

All methods are rerun from scratch using ten prespecified seeds that are disjoint from the three runs reported in the main tables. For known-K, methods with the same seed use identical observed categories, labeled supports, and data splits. Each seed controls the complete training run, including initialization, data order, augmentation, and method-specific stochastic operations. The baselines use their publicly released implementations and reported configurations. No published aggregate value enters this analysis. No hyperparameter or checkpoint is selected using the ten-run results or test annotations. Because this analysis uses new runs, its means need not equal the three-run averages in the main tables.

The ten-run mean and sample standard deviation are reported in Table 16. Under unknown-K, we include both the inferred cluster count $\widehat{K}$ and the absolute cluster-count error $|\widehat{K} - K|$. The reference K is used only after inference to compute this diagnostic error.

The complete training run, rather than an individual image, is treated as the statistical unit. For known-K, the shared category and support splits define matched runs, so we apply an exact two-sided paired permutation test to the ten seed-level differences. DeepDP-TGMM does not use labeled supports and therefore does not form meaningful pairs with DD-FSC. The unknown-K comparisons use an exact two-sided independent-sample permutation test.

For each of NMI, ACC, and ARI, let $\overline{m}$ denote the ten-run mean of the corresponding metric. We define $\Delta = \overline{m}_{\text{DD-FSC}} - \overline{m}_{\text{comp}}$. For cluster-count estimation, let $e = |\widehat{K} - K|$ and define $\Delta_K = \overline{e}_{\text{comp}} - \overline{e}_{\text{DD-FSC}}$. A positive value therefore consistently favors DD-FSC.

We estimate 95% bias-corrected and accelerated (BCa) bootstrap confidence intervals using 10,000 resamples. Known-K intervals resample the matched run pairs, whereas unknown-K intervals resample the two method groups independently. Holm correction controls the family-wise error rate. The known-K family contains the 12 comparisons formed by two datasets, two baselines, and three metrics. The unknown-K family contains the eight comparisons formed by two datasets, one baseline, and four metrics. The confidence intervals are unadjusted; all significance decisions therefore follow the Holm-adjusted p-values reported in Table 17.

Table 17: Statistical comparisons between DD-FSC and the strongest competing methods. The effect column reports Δ for clustering metrics and Δ_K for cluster-count error; positive values favor DD-FSC. Confidence intervals are 95% BCa intervals obtained from 10,000 bootstrap resamples. p_{Holm} denotes the permutation-test p-value after Holm correction within the corresponding known-K or unknown-K family.

(a) Known-K setting

Dataset	Comparator	Metric	Effect	95% CI	p_{Holm}
ImageNet-100	TIM++	NMI	-0.28	$[-0.51,\ 0.03]$	0.2813
		ACC	1.59	$[1.16,\ 2.05]$	0.0234
		ARI	6.33	$[5.74,\ 6.77]$	0.0234
	OSESSL	NMI	0.99	$[0.63,\ 1.35]$	0.0234
		ACC	1.13	$[0.83,\ 1.40]$	0.0234
		ARI	4.56	$[4.16,\ 4.86]$	0.0234
ImageNet-200	TIM++	NMI	1.19	$[0.86,\ 1.65]$	0.0234
		ACC	5.38	$[5.13,\ 5.60]$	0.0234
		ARI	1.82	$[1.41,\ 2.45]$	0.0234
	OSESSL	NMI	-0.28	$[-0.60,\ 0.22]$	0.2813
		ACC	4.27	$[3.77,\ 5.29]$	0.0234
		ARI	0.54	$[0.10,\ 1.16]$	0.2813

(b) Unknown-K setting

Dataset	Comparator	Metric	Effect	95% CI	p_{Holm}
ImageNet-100	DeepDP-TGMM	NMI	0.75	$[-0.04,\ 1.29]$	0.0973
		ACC	2.35	$[1.30,\ 3.17]$	0.0035
		ARI	-1.55	$[-2.44,\ -0.68]$	0.0277
		$\lvert\widehat{K} - K\rvert$	2.20	$[0.30,\ 3.80]$	0.0973
ImageNet-200	DeepDP-TGMM	NMI	2.53	$[1.60,\ 3.41]$	0.0009
		ACC	1.41	$[0.50,\ 2.25]$	0.0360
		ARI	1.98	$[0.68,\ 3.03]$	0.0352
		$\lvert\widehat{K} - K\rvert$	1.90	$[0.00,\ 4.40]$	0.1563

Under known-K, 9 of the 12 comparisons remain significant after Holm correction. On ImageNet-100, DD-FSC improves ACC and ARI over both baselines and NMI over OSESSL. On ImageNet-200, all three gains over TIM++ and the ACC gain over OSESSL are significant. The remaining NMI differences and the ImageNet-200 ARI difference against OSESSL are not significant.

Under unknown-K, 5 of the 8 corrected comparisons are significant. DD-FSC improves ACC on ImageNet-100 and NMI, ACC, and ARI on ImageNet-200. DeepDP-TGMM retains higher ARI on ImageNet-100, while the NMI difference there is not significant. The mean absolute cluster-count error decreases from 5.60 to 3.40 on ImageNet-100 and from 4.20 to 2.30 on ImageNet-200. Neither reduction remains significant after Holm correction ($p_{\text{Holm}} = 0.0973$ and 0.1563). Thus, the ten-run evidence supports repeatable assignment gains on both datasets, but not uniform superiority on every metric. The lower cluster-count errors are therefore directional findings rather than multiplicity-adjusted evidence of improvement.

Table 18: Backbone robustness under the known-K setting using OpenCLIP encoders. Results are averaged over three few-shot splits. Best results are in **bold** and second best are underlined.

Backbone	ImageNet-50			ImageNet-100			ImageNet-200			ImageNet-1K		
	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI
ViT-B/32	92.8	91.6	84.1	89.2	83.7	74.5	84.9	80.1	67.5	81.8	67.3	46.0
ViT-B/16	95.5	**93.4**	87.3	**92.4**	85.2	**77.4**	**88.7**	**83.7**	71.4	**86.4**	**72.4**	**51.2**
ViT-L/14	**95.8**	93.2	**87.5**	92.1	**85.4**	77.0	88.5	82.9	**71.5**	86.0	72.1	50.7

Table 19: Backbone robustness under the unknown-K setting using OpenCLIP encoders. Results are averaged over three few-shot splits. Best results are in **bold** and second best are underlined.

Backbone	ImageNet-50			ImageNet-100			ImageNet-200			ImageNet-1K		
	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI
ViT-B/32	78.7	73.6	62.9	68.2	53.0	40.9	59.1	42.4	30.8	54.2	31.9	23.6
ViT-B/16	**82.9**	**76.0**	66.8	69.7	**56.2**	**44.7**	**63.6**	**46.7**	**35.3**	**60.5**	**36.6**	**28.1**
ViT-L/14	82.0	74.5	**67.1**	**69.9**	55.6	42.1	63.4	46.0	33.6	59.1	35.3	26.9

Table 20: Cluster-count estimation under the unknown-K setting using different OpenCLIP ViT encoders. Results are reported as mean $\pm$ standard deviation over three few-shot splits. The ground-truth values are 50, 100, 200, and 1000, respectively. The closest estimate is in **bold**, and the second closest is underlined.

Backbone	ImageNet-50	ImageNet-100	ImageNet-200	ImageNet-1K
ViT-B/32	48.0 ± 1.4	109.3 ± 2.6	193.7 ± 3.4	$\mathbf{1048.7 \pm 14.8}$
ViT-B/16	51.7 ± 0.9	$\mathbf{103.0 \pm 1.4}$	$\mathbf{198.7 \pm 1.9}$	1049.0 ± 10.2
ViT-L/14	$\mathbf{51.0 \pm 2.4}$	96.0 ± 0.8	204.7 ± 2.6	1054.0 ± 13.4

G Robustness Analysis

G.1 OpenCLIP Backbone Variants

DD-FSC operates on features extracted by a frozen pretrained image encoder. We evaluate three models from the OpenCLIP family: ViT-B/32, ViT-B/16, and ViT-L/14. They use the publicly released `laion2b_s34b_b79k`, `laion2b_s34b_b88k`, and `laion2b_s32b_b82k` checkpoints, respectively. This within-family comparison changes the patch resolution and model capacity while retaining the same pretraining objective.

All backbones remain frozen throughout training. ViT-B/32 and ViT-B/16 produce 512-dimensional features, whereas ViT-L/14 produces 768-dimensional features. Each encoder uses its native normalized embedding and preprocessing pipeline. Only dimension-dependent layers in the adapter, score network, prototypes, and DP model are resized. The optimization schedule, objective weights, data splits, and all other dataset-specific settings remain unchanged.

We follow the partial-category five-shot protocol on ImageNet-50, ImageNet-100, ImageNet-200, and ImageNet-1K. Every configuration uses the same three few-shot splits. Known-K results are reported in Table 18. Unknown-K performance and the corresponding cluster-count estimates are reported in Tables 19 and 20, respectively.

Tables 18 and 19 compare assignment quality across the three encoders. Under known-K, ViT-B/16 ranks first on eight of the twelve metric–dataset pairs, while ViT-L/14 leads on the remaining four. Under unknown-K, ViT-B/16 ranks first on ten pairs; ViT-L/14 obtains the best ARI on ImageNet-50 and the best NMI on ImageNet-100. Thus, increasing encoder capacity does not produce a uniform improvement. ViT-B/32 gives lower scores, with larger gaps on ImageNet-200 and ImageNet-1K, but remains stable across the three few-shot splits. In particular, every reported standard deviation is at most 1.5 points. These results show that DD-FSC remains effective across the tested feature dimensions and resolutions, while sufficiently resolved representations remain important for larger category spaces.

Table 20 further evaluates whether this robustness extends to adaptive partition recovery. The closest estimate is obtained by ViT-L/14 on ImageNet-50, ViT-B/16 on ImageNet-100 and ImageNet-200,

Table 21: Supervision-robustness protocol on ImageNet-100 under known-K. Each factor is varied independently while all remaining settings are fixed. The uniform three-shot configuration is the reference for the equal-budget allocation test.

Setting	Factor	Tested values	Default/reference
Known-K	Shots/category q	$\{1, 3, 5, 10, 15\}$	5
	Category coverage ρ_l	$\{10, 20, 30, 40, 50, 60\}\%$	30%
	Label-noise ratio η_l	$\{0, 10, 20, 30\}\%$	0%
	Support allocation	Uniform three-shot, imbalanced 1/3/5-shot	Uniform

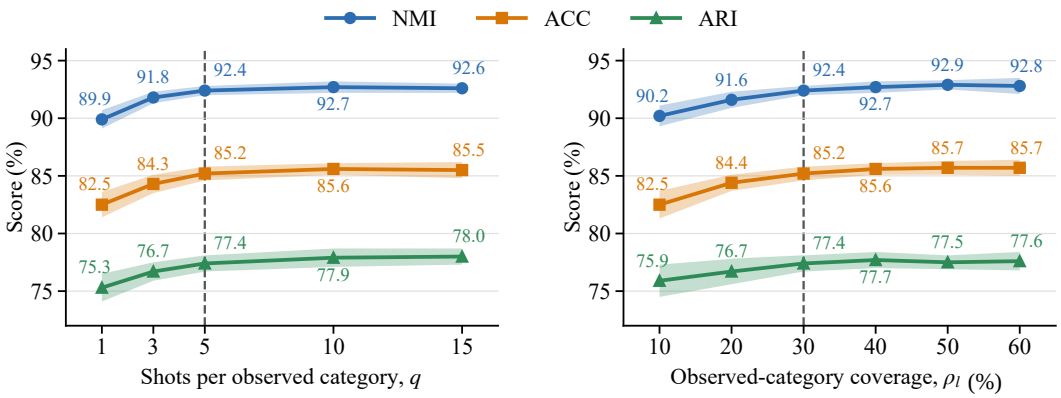


Figure 4: Annotation-quantity sensitivity on ImageNet-100 under known-K. Panels (a) and (b) vary the shots per observed category and the observed-category coverage, respectively. Curves and shaded regions show the mean and one standard deviation over three prespecified seeds. Dashed lines mark the default configuration.

and ViT-B/32 on ImageNet-1K. Across all backbone–dataset pairs, the relative cluster-count error is at most 9.3%. No encoder therefore induces a uniform over- or under-segmentation bias. Together with the assignment results, these estimates support the backbone robustness of the bidirectional design: density-guided refinement and DP inference remain coordinated as the frozen feature geometry changes. This behavior is obtained without backbone-specific optimization or benchmark-specific encoder selection within the tested OpenCLIP family.

G.2 Few-Shot Supervision Conditions

We further examine whether DD-FSC remains stable as the amount, coverage, reliability, and allocation of few-shot supervision change. All experiments use ImageNet-100 under known-K, with the same frozen backbone, optimization schedule, and three prespecified seeds as the main experiments. We vary one annotation factor at a time while holding all remaining settings fixed to those reported in Appendix C.2. Curves report the mean over three runs, shaded regions represent one standard deviation, and vertical dashed lines mark the default values used in the main experiments.

Table 21 summarizes the supervision variations. Annotation quantity is controlled by the number of labeled samples per observed category and the fraction of categories containing labeled supports. Annotation reliability is evaluated through controlled label corruption and nonuniform per-category allocation.

Figure 4 examines the two dimensions that define the partial-category few-shot protocol. Panel (a) varies the number of labeled samples per observed category while fixing category coverage at 30%. Panel (b) varies the observed-category coverage while fixing the annotation level at five shots per category. NMI, ACC, and ARI are plotted on a common vertical scale so that their responses to additional supervision can be compared directly.

Increasing the annotation level from one to five shots improves NMI, ACC, and ARI by 2.5, 2.7, and 2.1 points, respectively. Increasing observed-category coverage from 10% to 30% yields corresponding gains of 2.2, 2.7, and 1.5 points. Beyond the default configuration, the largest changes are only 0.6 points for the shot sweep and 0.5 points for the coverage sweep. Both sweeps therefore exhibit diminishing returns once sufficient semantic anchors are available. This pattern is consistent

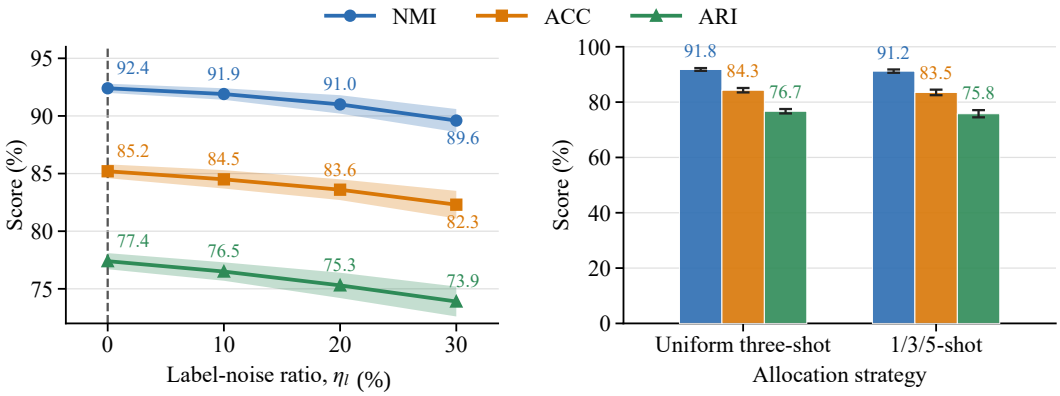


Figure 5: Annotation-quality and allocation sensitivity on ImageNet-100 under known-K. Panel (a) corrupts the labeled supports. Panel (b) compares uniform and imbalanced allocations under the same 90-label budget. Results are the mean and one standard deviation over three prespecified seeds.

with density-guided expansion using sparse annotations effectively without depending on a larger labeling budget.

Annotation quantity alone does not capture the reliability or distribution of supervision. Figure 5 therefore evaluates controlled label corruption and a fixed-budget allocation shift. For the corruption experiment, an η_l fraction of the labeled supports is assigned uniformly to incorrect observed categories while the images, support locations, and unlabeled pool remain unchanged. For the allocation experiment, the 30 observed categories are divided equally among one-, three-, and five-shot groups. The uniform three-shot reference has the same total budget of 90 labels, isolating allocation from annotation quantity.

At label-noise ratios of 10%, 20%, and 30%, ACC decreases by 0.7, 1.6, and 2.9 points relative to clean supervision. The corresponding NMI decreases are 0.5, 1.4, and 2.8 points, while ARI decreases by 0.9, 2.1, and 3.5 points. The monotonic degradation confirms the expected effect of corrupted supervision, while remaining gradual under moderate noise. Under the same 90-label budget, the imbalanced 1/3/5-shot allocation reduces NMI, ACC, and ARI by only 0.6, 0.8, and 0.9 points. The gradual noise response and sub-one-point allocation gaps indicate that semantic expansion tolerates the tested moderate corruption and does not depend strongly on equal support counts.

H Additional Hyperparameter Sensitivity Analysis

Section 4.4 evaluates the three core hyperparameters that control semantic expansion and DP–diffusion coupling: the confidence threshold δ, candidate budget P, and DP-guidance weight λ_{DP}. We extend that analysis to additional method-specific hyperparameters governing score-guided exploration, representation objectives, and adaptive inference.

All experiments are conducted on ImageNet-100 using the same frozen backbone, optimization schedule, and three prespecified seeds as the main experiments. We vary one factor at a time while holding all remaining settings fixed to those reported in Appendix C.2. The default values were fixed before the sensitivity sweeps and were not selected using the reported test results. Score-guidance and representation-objective experiments use the **known-K** setting to isolate their effects from cluster-count estimation. Parameters governing posterior activation and truncation are evaluated separately under **unknown-K**. Curves report the mean over three runs, shaded regions represent one standard deviation, and vertical dashed lines mark the default values used in the main experiments.

Table 22 summarizes the additional hyperparameter sweeps. The Langevin base step coefficient ϵ_0 and exploration scale γ control the deterministic score drift and stochastic exploration, respectively. The score weight λ_{score} controls score-network learning, while the FCL, DGCL, and PCL weights balance the three representation objectives. Under unknown-K, the normalized-ELBO

Table 22: Additional hyperparameter-sensitivity protocol on ImageNet-100. Each factor is varied independently while all remaining settings are fixed. Default values are those used in the main experiments. The sweeps of δ, P, and λ_{DP} are reported in Section 4.4.

Setting	Factor	Tested values	Default
Known-K	Base step coefficient ϵ_0	$\{0.0025, 0.005, 0.01, 0.02, 0.04\}$	0.01
	Exploration scale γ	$\{0, 0.0025, 0.005, 0.0075, 0.01\}$	0.005
	Score weight λ_{score}	$\{0.0125, 0.025, 0.05, 0.1, 0.2\}$	0.05
	FCL weight λ_{FCL}	$\{0.005, 0.01, 0.02, 0.04, 0.08\}$	0.02
	DGCL weight λ_{DGCL}	$\{0.0025, 0.005, 0.01, 0.02, 0.04\}$	0.01
	PCL weight λ_{PCL}	$\{0.00125, 0.0025, 0.005, 0.01, 0.02\}$	0.005
Unknown-K	Normalized ELBO weight λ_{ELBO}	$\{0, 0.125, 0.25, 0.5, 1, 2\}$	0.5
	Active threshold ε	$\{0.0025, 0.00375, 0.005, 0.0075, 0.01\}$	0.005
	Truncation capacity $K_{\max}$	$\{100, 150, 200, 300, 400\}$	200

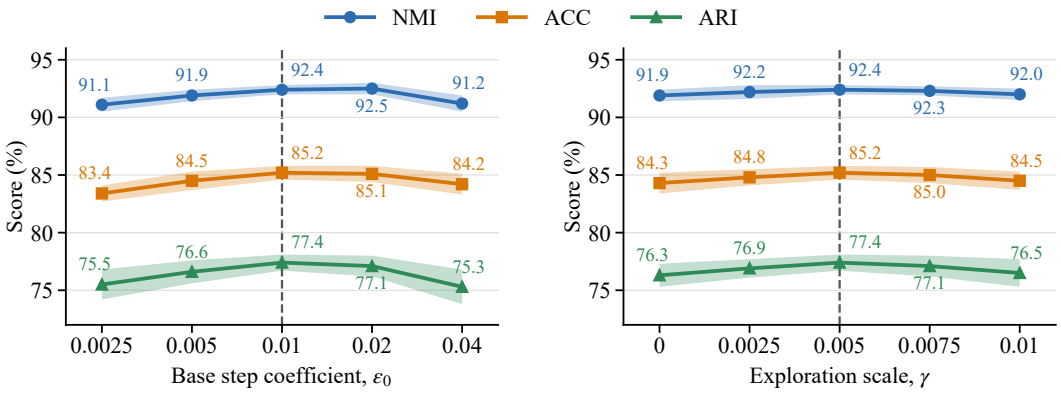


Figure 6: Sensitivity to score-guided exploration controls on ImageNet-100 under known-K. Panels (a) and (b) vary the base score-step coefficient and Gaussian exploration scale, respectively. Curves and shaded regions show the mean and one standard deviation over three prespecified seeds.

weight λ_{ELBO} controls the direct variational feedback to the encoder. The active threshold ε controls posterior component activation, while $K_{\max}$ determines the capacity of the truncated variational approximation. We focus on method-specific controls rather than generic optimizer settings, which are reported in Appendix C.2.

H.1 Score-Guided Exploration Controls

The density-guided update uses $\epsilon_\ell = \epsilon_0(\sigma_\ell/0.15)^2$, where ϵ_0 sets the deterministic score-step scale. Its stochastic term is $\gamma\sqrt{2\epsilon_\ell}z$, where γ controls the magnitude of Gaussian exploration. Figure 6 varies these two parameters independently under known-K. This analysis complements Theorem 1, whose finite-step density-ascent condition depends explicitly on the step size, while separately examining the stochastic exploration term that does not enter the conditional mean drift.

Across the tested base step coefficients, NMI, ACC, and ARI vary by 1.4, 1.8, and 2.1 points, respectively. Varying the exploration scale produces corresponding ranges of 0.5, 0.9, and 1.1 points. Relative to the deterministic setting $\gamma = 0$, the default exploration scale improves the three metrics by 0.5, 0.9, and 1.1 points. Performance is stable for $\epsilon_0 \in [0.005, 0.02]$, whereas the two extremes are less effective. This behavior is consistent with insufficient density movement at small steps and overly large finite-step updates at the other extreme. The strong deterministic result indicates that score drift as the primary signal. Moderate stochastic exploration provides an additional gain, but larger noise reverses part of that gain.

H.2 Representation Objective Weights

The ablations in Table 4 evaluate whether FCL, DGCL, and PCL are necessary. Figure 7 complements those removal tests by varying the nonzero coefficient of each representation objective around

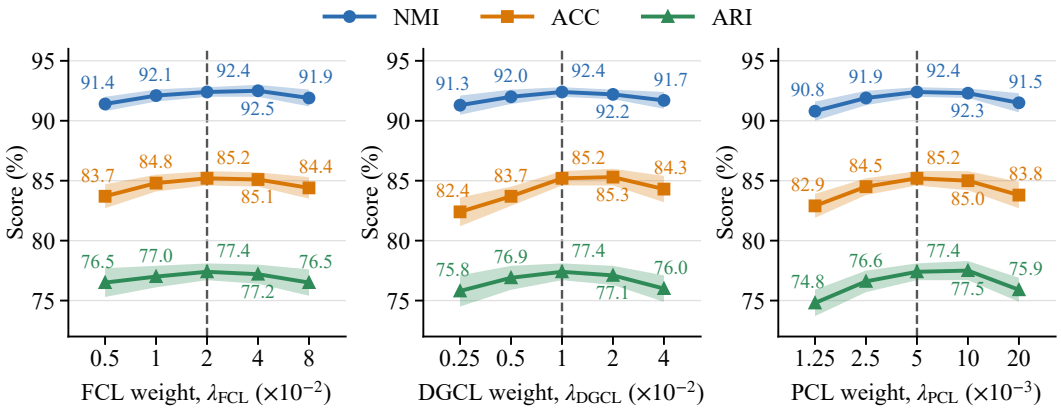


Figure 7: Sensitivity of DD-FSC to the FCL, DGCL, and PCL objective weights on ImageNet-100 under known-K. Each coefficient is varied independently around its default. Curves and shaded regions show the mean and one standard deviation over three prespecified seeds. Dashed lines mark the defaults.

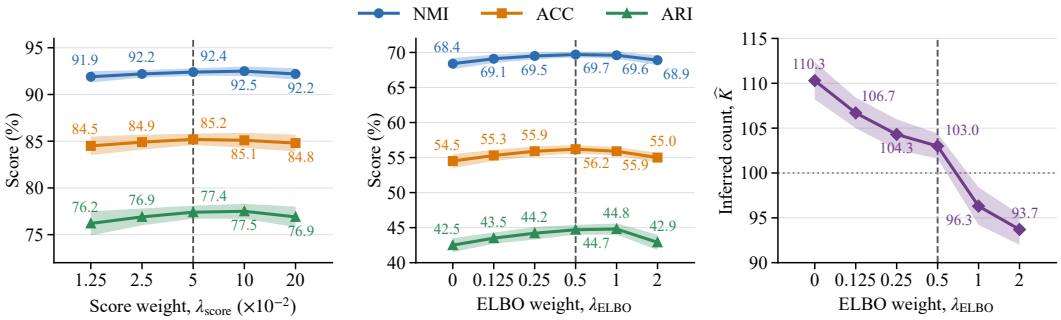


Figure 8: Sensitivity to the score and normalized-ELBO weights on ImageNet-100. Panel (a) evaluates λ_{score} under known-K. Panels (b) and (c) report clustering performance and the inferred cluster count when varying λ_{ELBO} under unknown-K. Curves and shaded regions denote the mean and one standard deviation over three prespecified seeds. Vertical dashed lines indicate the default values, and the horizontal dotted line marks the evaluation reference at $K = 100$.

its default value. The three sweeps use multiplicative factors of $\{0.25, 0.5, 1, 2, 4\}$ and are conducted under known-K so that changes in representation learning are not confounded by cluster-count estimation.

Across the FCL sweep, NMI, ACC, and ARI vary by 1.1, 1.5, and 0.9 points, respectively. The corresponding variations are 1.1, 2.9, and 1.6 points for DGCL, and 1.6, 2.3, and 2.7 points for PCL. Across all three sweeps, the default configuration is within 0.1 point of the best tested value for each metric. The removal tests in Table 4 verify the contribution of each objective, whereas these nonzero sweeps reveal a broad useful weighting range. The larger ACC and ARI changes for DGCL and PCL are consistent with their direct roles in density-aware feature refinement and prototype-level separation. The overlapping uncertainty bands near the defaults indicate that exact coefficient tuning is unnecessary.

Beyond these three representation objectives, the total objective also contains coefficients for score estimation and normalized variational feedback. Figure 8 varies λ_{score} under known-K and λ_{ELBO} under unknown-K. The former controls score-network learning, whereas the latter controls the direct contribution of the normalized ELBO to encoder learning. The CAVI updates remain unchanged when λ_{ELBO} is varied.

Across the tested score weights, NMI, ACC, and ARI vary by 0.6, 0.7, and 1.3 points, respectively. The default setting achieves the best ACC and remains within 0.1 points of the best NMI and ARI. Across the nonzero normalized-ELBO weights, the corresponding variations are 0.8, 1.2, and 1.9 points, while $\widehat{K}$ ranges from 93.7 to 106.7. Relative to $\lambda_{\text{ELBO}} = 0$, the default value improves NMI, ACC, and ARI by 1.3, 1.7, and 2.2 points and reduces the absolute cluster-count error from 10.3 to

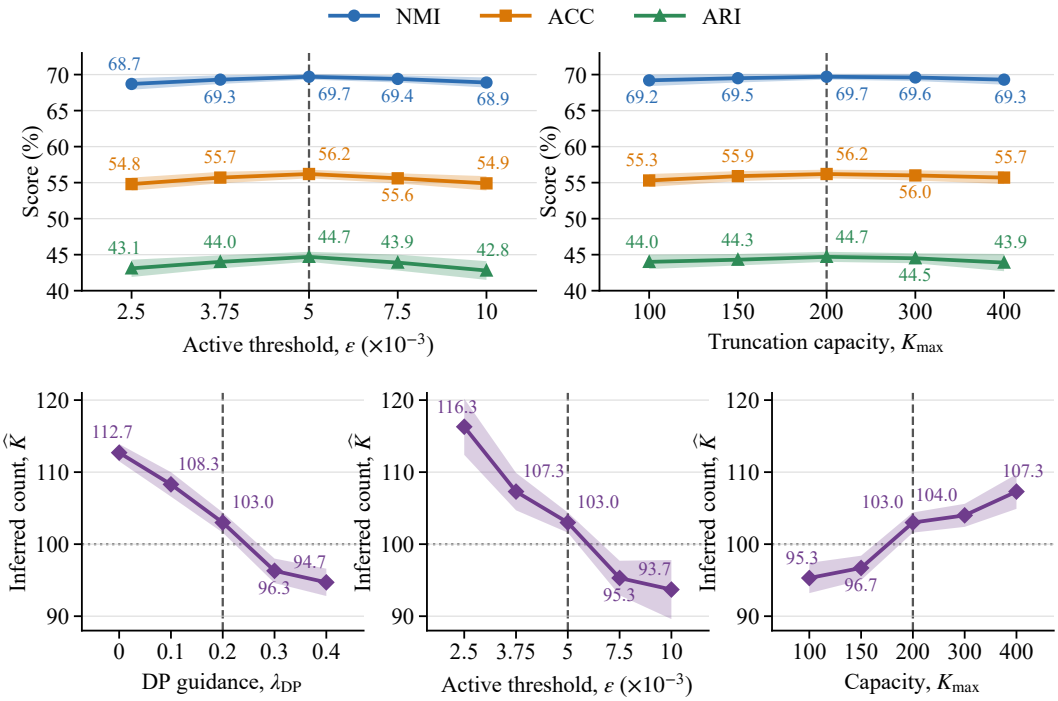


Figure 9: Sensitivity of adaptive cluster inference on ImageNet-100 under unknown-K. The top row reports clustering performance. The bottom row reports the inferred cluster count. Curves and shaded regions show the mean and one standard deviation over three prespecified seeds. The horizontal line at $K = 100$ is used only for post-inference evaluation.

3.0. The score term is therefore stable around its default. The ELBO sweep is consistent with coordinated representation and partition learning: moderate variational feedback improves both cluster assignments and category-count recovery, whereas excessive feedback leads to under-segmentation.

H.3 Adaptive Cluster Discovery Controls

Section 4.4 reports clustering performance across the DP-guidance weights. We additionally examine how λ_{DP} affects the inferred cluster count and evaluate the two parameters that control posterior activation and variational capacity. Figure 9 varies the active-component threshold ε and truncation capacity $K_{\max}$ under unknown-K. These are one-factor sweeps: $K_{\max}$ is fixed to 200 when ε varies, and ε is fixed to 0.005 when $K_{\max}$ varies. The horizontal line at $K = 100$ and $|\widehat{K} - K|$ are used only for post-inference evaluation; the role of K_{ref} in setting the default capacity ceiling is stated explicitly in Appendix C.2.

Across the tested DP-guidance weights, $\widehat{K}$ ranges from 94.7 to 112.7. The count moves from over-segmentation to under-segmentation as the coupling strength increases. Varying ε changes NMI, ACC, and ARI by 1.0, 1.4, and 1.9 points, respectively, with $\widehat{K}$ ranging from 93.7 to 116.3. The threshold sweep exposes the expected transition from retaining low-mass components to suppressing valid ones. Across the tested truncation capacities, the metric variations are only 0.5, 0.9, and 0.8 points, while $\widehat{K}$ ranges from 95.3 to 107.3. For $K_{\max} \geq 150$, the inferred count remains well below the nominal capacity. The default settings yield $\widehat{K} = 103.0 \pm 1.4$, corresponding to an absolute error of 3.0. Within this ImageNet-100 sweep, performance is stable once the capacity reaches 150, while posterior activation produces an interior $\widehat{K}$. This supports robustness to the tested ceiling values; $K_{\max}$ itself is not treated as the cluster-count estimate.

Overall, the main-text and appendix sweeps separate four potential sources of sensitivity. The confidence threshold and candidate budget control how sparse supervision admits unlabeled candidates. The score-step coefficient and exploration scale govern density-guided trajectories. The score and representation-objective weights balance density estimation and representation learning.

The normalized-ELBO weight controls direct variational feedback to the encoder, while the remaining DP parameters govern the evolving nonparametric partition. Across all tested factors, the largest metric variation is 8.4 points, and the largest variation in $\widehat{K}$ is 22.6 clusters. Most parameters exhibit stable neighborhoods around their defaults. The larger changes occur when DP guidance is removed or posterior activation is driven toward over- or under-segmentation. This pattern distinguishes ordinary tuning tolerance from the structural controls that govern adaptive cluster discovery.

I Visualization Analysis

The quantitative results establish the effectiveness of DD-FSC under both known- and unknown-K settings. We complement them with two types of t-SNE analysis, each conducted under both settings. We first compare DD-FSC with representative competing methods. We then examine how its learned representation differs from the shared initial OpenCLIP feature space across four dataset scales. This sequence moves from external comparison to cross-scale representation evolution.

All t-SNE visualizations use the complete test set of the corresponding dataset without category or sample subsampling. Within each dataset, the same test images, sample order, and category-color mapping are retained across all compared representations and methods. Ground-truth labels are used only for post-hoc point coloring. They do not affect training, cluster inference, checkpoint selection, or hyperparameter selection.

The known-K and unknown-K results are obtained from separate setting-specific training runs. For a given dataset, both settings use the same frozen OpenCLIP ViT-B/16 encoder, deterministic preprocessing, test images, and initial normalized features. Their initial OpenCLIP projections are therefore identical. In contrast, the final panels use the optimized adapter outputs from the corresponding setting-specific runs. These representations may differ because the DP truncation, active-component rule, posterior responsibilities, diffusion guidance, and optimization configuration depend on whether K is provided.

Unless otherwise specified, t-SNE uses perplexity 30, learning rate 200, PCA initialization, 1,000 iterations, and seed 42. Final representations from independently trained models are projected separately using identical t-SNE settings. We compare qualitative patterns of local category mixing and neighborhood organization, not absolute coordinates, inter-cluster distances, or cluster sizes in the two-dimensional space. In particular, the number of visually separated groups is not interpreted as an estimate of K. The projections are not used for model selection, hyperparameter selection, or quantitative evaluation. The random seed controls only the t-SNE initialization and does not affect the set of visualized samples.

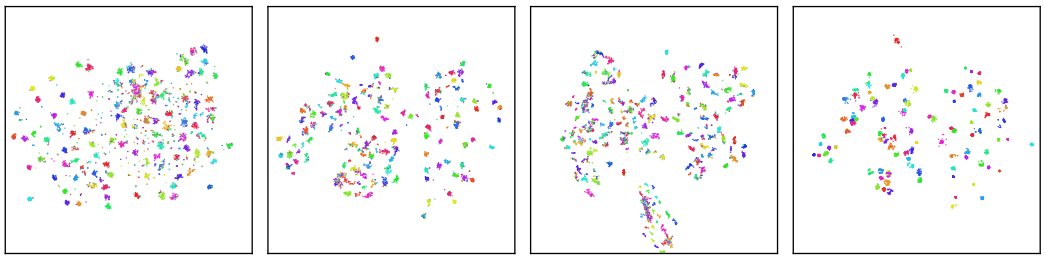
I.1 Comparison with Representative Methods

We first compare the final representations of DD-FSC and representative methods from the quantitative evaluation. Under known-K, the comparison includes cluster-FSL, TIM++, and OSESSL. Under unknown-K, it includes DeepDPM, SCMax, and DeepDP-TGMM. All methods use the same ImageNet-100 test samples and annotation protocol.

Figures 10 and 11 visualize the representations used by the final clustering or assignment procedure. We compare local compactness, category mixing, and the separation between nearby groups. The projections provide qualitative evidence; the main tables remain the basis for quantitative comparison.

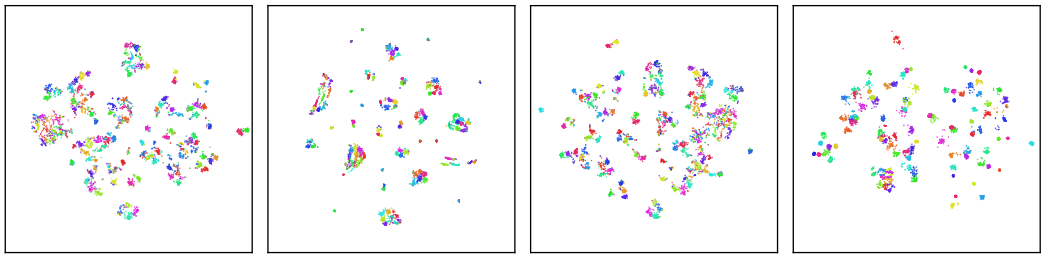
As shown in Figure 10, cluster-FSL and TIM++ retain broad, intertwined manifolds with substantial local category mixing. OSESSL separates more local groups, but several regions remain diffuse. DD-FSC instead forms smaller and more distinct local modes, with less interleaving between neighboring categories. This structure is consistent with density-guided learning consolidating sparse semantic evidence around high-density regions rather than preserving the original neighborhood geometry.

Figure 11 exhibits a similar qualitative contrast when the cluster count is not provided. DeepDPM and SCMax retain extended regions with considerable local category mixing. DeepDP-TGMM recovers several localized groups, although adjacent categories remain connected in crowded regions. The DD-FSC projection contains more localized neighborhoods and less visible interleaving among



(a) cluster-FSL (b) TIM++ (c) OSESSL (d) DD-FSC (Ours)

Figure 10: t-SNE comparison with representative methods under known-K on ImageNet-100. The panels visualize the representations used for final assignment by cluster-FSL, TIM++, OSESSL, and DD-FSC. Each panel contains the complete test set without subsampling. Colors denote ground-truth categories and are used only for post-hoc visualization.



(a) DeepDPM (b) SCMax (c) DeepDP-TGMM (d) DD-FSC (Ours)

Figure 11: t-SNE comparison with representative methods under unknown-K on ImageNet-100. The panels visualize the representations used for final clustering by DeepDPM, SCMax, DeepDP-TGMM, and DD-FSC. Each panel contains the complete test set without subsampling. Colors denote ground-truth categories and are used only after inference for visualization.

several nearby categories. Together with the unknown-K clustering and cluster-count results, this pattern is consistent with density-guided representation learning remaining effective while the DP posterior adapts the active partition.

I.2 Evolution of Representation Structure

Having compared the final representations with representative methods, we next examine the difference between the initial OpenCLIP feature space and the representation learned by DD-FSC across datasets of different scales. The initial representation is the ℓ_2-normalized output of the frozen OpenCLIP ViT-B/16 image encoder. Since the residual adapter is identity-initialized, the same feature is obtained at the start of both known-K and unknown-K training. The final representation is the normalized adapter output from the corresponding setting-specific checkpoint. We compare these states on ImageNet-50, ImageNet-100, ImageNet-200, and ImageNet-1K.

For a given dataset, the initial OpenCLIP panel is identical across the known-K and unknown-K figures. The final panels are generated from their respective optimized representations. Within every comparison, the same images and category colors are retained. Because the final t-SNE projections are fitted separately, we interpret only qualitative changes in local category mixing and neighborhood organization. We do not interpret point-wise displacement, absolute cluster size, or the number of visible two-dimensional groups.

Figure 12 shows that the frozen OpenCLIP features already contain semantic neighborhoods, but many categories occupy broad and overlapping regions. After DD-FSC optimization, ImageNet-50 and ImageNet-100 exhibit markedly tighter local modes with clearer gaps between nearby groups. On ImageNet-200, large mixed regions are reorganized into more localized neighborhoods. The ImageNet-1K projection remains crowded; at this scale, the change is expressed through local neighborhood structure rather than globally isolated clusters. This scale-dependent pattern is consistent with DD-FSC consolidating local density structure rather than relying on globally separated backbone features.

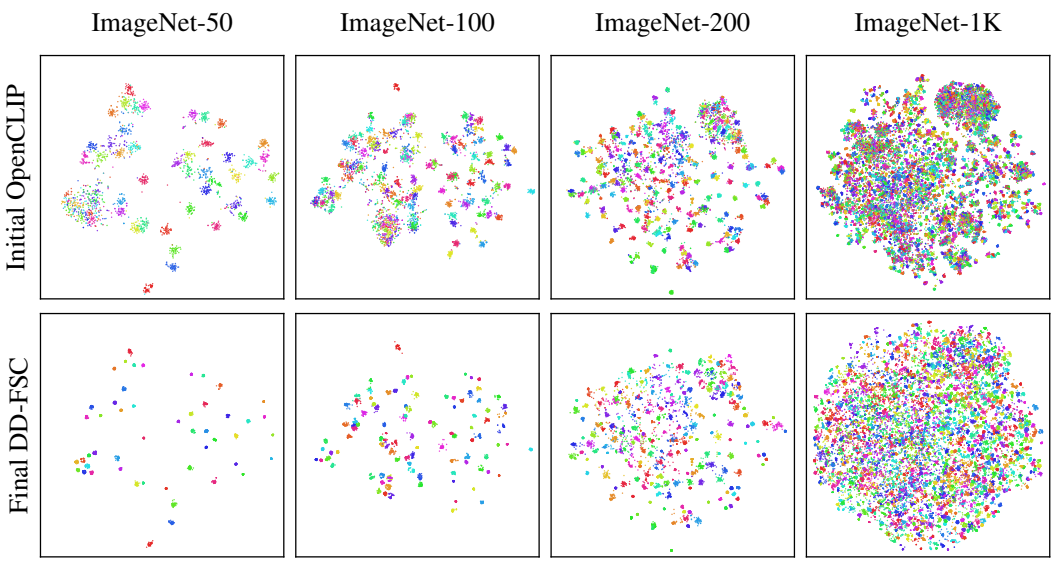


Figure 12: Evolution from the initial OpenCLIP representation to the final DD-FSC representation under known-K. The top and bottom rows show the initial and final representations, respectively. Columns correspond to ImageNet-50, ImageNet-100, ImageNet-200, and ImageNet-1K. Every panel contains the complete test set of the corresponding dataset. Colors denote ground-truth categories and are used only for post-hoc visualization.

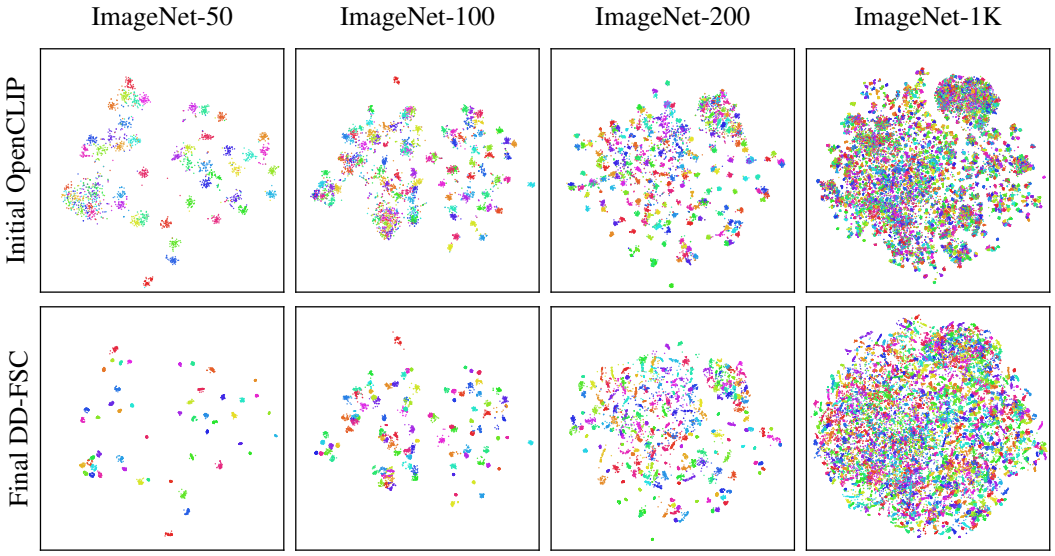


Figure 13: Evolution from the initial OpenCLIP representation to the final DD-FSC representation under unknown-K. The top and bottom rows show the initial and final representations, respectively. Columns correspond to ImageNet-50, ImageNet-100, ImageNet-200, and ImageNet-1K. Every panel contains the complete test set of the corresponding dataset. Colors denote ground-truth categories and are used only after inference for visualization.

Figure 13 shows a qualitatively similar transition when the cluster count is not provided. The difference is most visible on ImageNet-50 and ImageNet-100, where the final projections contain more localized category neighborhoods and less visible mixing in several regions. ImageNet-200 and ImageNet-1K remain globally crowded as the category scale increases, although local category organization is still visible after optimization. Together with the quantitative unknown-K results, this recurrence indicates that density-guided representation refinement remains effective when the active partition and its cardinality are inferred by the DP posterior rather than supplied in advance.