CAGS: CLASS-ADAPTIVE GUIDANCE STRENGTH FOR TRAINING-FREE DIFFUSION-BASED DATASET DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Training-free diffusion-based dataset distillation methods such as MGD^3 typically use a single classifier-free guidance (CFG) strength for all classes. This class-agnostic design overlooks substantial variation in class complexity: structurally diverse classes may be under-guided, whereas simpler classes can suffer from excessive sharpening. Our key insight is that class-wise generation statistics can reveal how strongly each class should be guided. Based on this observation, we propose CAGS (Class-Adaptive Guidance Strength), a training-free method that assigns a class-specific CFG strength using cluster entropy and intra-class variance. A systematic ablation and weight search over an initial four-factor design shows that these two factors consistently drive the gains, while mode count and inter-class separability can be removed. We further provide a theoretical explanation under a Gaussian-mixture model, where the optimal guidance satisfies $\lambda_c^* \propto v_c/\sigma_c^2$ and increases with both intra-class variance and entropy; consequently, the benefit of adaptation grows with cross-class heterogeneity. On ImageNet-100 at IPC=10, CAGS achieves 30.81% on ResNet-18, 29.74% on ResNetAP-10, and 25.55% on ConvNet-6, improving over MGD^3 by 7.21, 3.94, and 2.15 percentage points, respectively, and outperforming other published baselines including IDC-1 and MinMaxDiff. In matched scaling experiments, the gains increase to 3.58–6.97 points on ImageNet-1000, while remaining near-neutral on a low-heterogeneity 10-class subset and reaching +5.28 points on ImageNette. CAGS also generalizes across generators, improving Stable Diffusion 1.5 results on Food-101 from 2.27% to 4.17%, with zero trainable parameters and negligible inference overhead. Code is available at `https://anonymous.4open.science/r/AGS-DD-1B3D/`.

1 INTRODUCTION

Dataset distillation condenses large training sets into compact synthetic surrogates that preserve training utility. Gradient matching (Zhao et al., 2023), trajectory matching (Cazenavette et al., 2022), and distribution matching (Liu et al., 2023) typically require repeated optimization or model training, limiting scalability. SRe^{2}L (Yin et al., 2023) reduces dependence on bi-level optimization, while MGD^3 (Chan-Santiago et al., 2025) directly samples from a frozen class-conditional Diffusion Transformer (DiT) (Peebles & Xie, 2023). This training-free approach makes the sampling strategy central to distilled-data quality.

However, MGD^3 uses one classifier-free guidance (CFG) (Ho & Salimans, 2022) strength for all classes. CFG trades conditional fidelity against diversity, but classes differ in structure and variability: stronger guidance may help represent a diverse class while over-constraining a simpler one. A global strength therefore ignores class-specific guidance requirements.

We view this mismatch as a *class-allocation problem*. Intra-class variance measures sample spread, while cluster entropy measures how broadly probability mass is distributed across internal modes; together, they estimate how difficult each class is to represent. Under a Gaussian-mixture model, our analysis connects optimal guidance to these statistics and predicts greater benefits from adaptation as cross-class heterogeneity increases.

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.

We propose AGS-DD (Adaptive Guidance Allocation for Diffusion Dataset Distillation), centered on CAGS (Class-Adaptive Guidance Strength), which assigns per-class CFG strengths from cluster entropy and intra-class variance. With the DiT frozen, CAGS operates entirely at inference time, adds no trainable parameters, and preserves MGD^3's training-free, single-pass sampling. Adaptation along the IPC and diffusion-timestep axes yields consistently weaker results, identifying class-wise adaptation as the effective component.

We evaluate AGS-DD on ImageNet-100, ImageNette, ImageWoof, a 200-class ImageNet subset, ImageNet-1K, Food-101, and CIFAR-10, using ConvNet-6, ResNet-18, and ResNetAP-10 under multiple IPC budgets. On ImageNet-100 at IPC=10, CAGS achieves 30.81% top-1 accuracy with ResNet-18 versus MGD^3's 23.6% ($+7.21$ percentage points), also surpassing reported baselines on ResNetAP-10 and ConvNet-6. Relative gains reach 31–44% on ImageNet-1K. Heterogeneity matters even at the same class count: a low-heterogeneity 10-class ImageNet subset shows near-neutral results, whereas ImageNette gains 5.28 points. With Stable Diffusion 1.5 on Food-101, accuracy improves from 2.27% to 4.17%, demonstrating transfer across generators. Our contributions are threefold:

- **We identify class-wise guidance mismatch as a key limitation of training-free diffusion dataset distillation.** We show that the effectiveness of adaptive guidance is governed by cross-class heterogeneity rather than class count alone, and provide a theoretical analysis explaining why greater heterogeneity creates a larger adaptivity gap.
- **We introduce AGS-DD, centered on the training-free CAGS mechanism for class-adaptive CFG.** CAGS derives per-class guidance strengths from cluster entropy and intra-class variance, requires zero additional training or trainable parameters, and preserves the inference-only sampling pipeline of existing diffusion-based distillation.
- **We validate consistent improvements across datasets, architectures, scales, and generators.** CAGS improves ResNet-18 accuracy on ImageNet-100 from 23.6% to 30.81%, yields increasingly large gains at ImageNet-1K scale, and transfers to Stable Diffusion 1.5 on Food-101, while extensive ablations confirm that class-wise adaptation is the effective source of improvement.

2 Related Work

2.1 Dataset Distillation

Dataset distillation synthesizes a compact dataset $\mathcal{S}$ for training models to approximate performance on the full dataset $\mathcal{T}$. Gradient matching (Zhao et al., 2023), trajectory matching (Cazenavette et al., 2022), and distribution matching (Liu et al., 2023) jointly optimize synthetic data and a surrogate model in a bi-level loop. Effective on CIFAR and SVHN, their cost limits ImageNet scalability (Russakovsky et al., 2015). SRe^{2}L (Yin et al., 2023) decoupled optimization into generative-model training, image synthesis, and classifier training. IDC (Kim et al., 2022) improved synthetic-data parameterization through differentiable augmentation, while DD-Ranking (Li et al., 2025b) standardized evaluation and showed that many methods struggle to outperform real-data subsampling.

2.2 Generative Dataset Distillation with Diffusion Models

Pretrained diffusion models (Ho et al., 2020; Peebles & Xie, 2023) enable generative distillation. D^{4}M (Su et al., 2024) disentangles label- and image-conditional generation using Stable Diffusion, with VAE encoding and partial noising of real images before denoising. MGD^3 (Chan-Santiago et al., 2025) directly samples from a frozen class-conditional DiT-XL/2, enabling training-free distillation. Other approaches target text-to-image generation (CoDA (Zhou et al., 2026)), architectural-prior decoupling (PRISM (Moser et al., 2026)), curriculum-based sample selection with iterative classifier training (LGD (Chan-Santiago & Shah, 2026)), optimal-transport distribution matching (DMGD (Wang et al., 2026)), semantic-aware latent selection (SAS (Li et al., 2026b)), hierarchical manifold guidance (ManifoldGD (Roy et al., 2026)), and allocation-aware distillation (Pool-Select-Refine (Li et al., 2026c)). Our per-class adaptation replaces MGD^3's fixed guidance, remains training-free, and is orthogonal to distribution-matching (DMGD), sample-selection (LGD), and latent-selection (SAS) approaches.

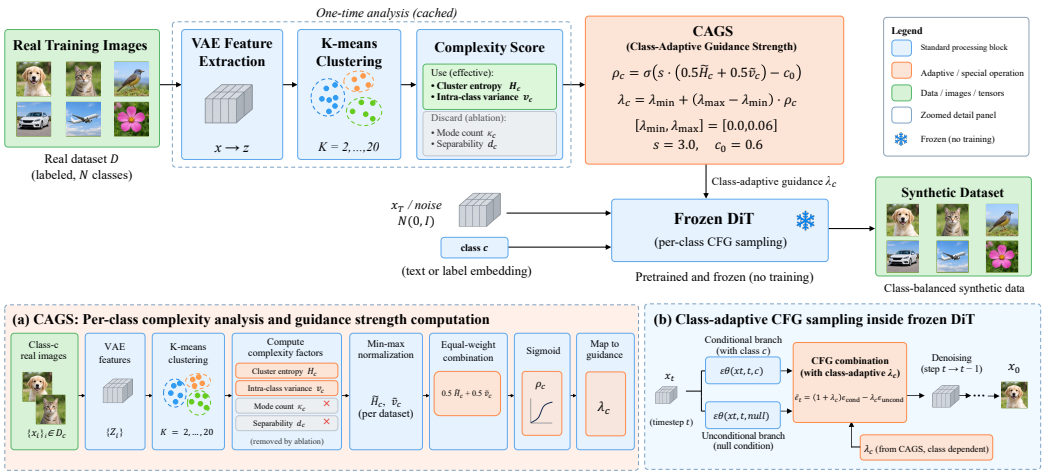

Figure 1: Overview of CAGS. The real dataset is analyzed once to estimate per-class complexity from cluster entropy and intra-class variance, which is mapped to a class-adaptive guidance strength λ_c. The resulting guidance is used for classifier-free sampling with a frozen pretrained DiT, requiring no model training.

2.3 Classifier-Free Guidance and Adaptive Guidance Schedules

CFG (Ho & Salimans, 2022) trades diversity for conditioning fidelity as inference-time guidance strength λ increases. Adaptive approaches include decreasing λ during denoising (Yehezkel et al., 2025), modulation by conditioning complexity (Zhang & Wan, 2025), timestep- and conditioning-dependent scaling (Malarz et al., 2025), and dynamic low-confidence masking to identify regions where guidance is most needed (Li et al., 2025a). Other work studies compact model calibration via dual-branch score distillation (Shafi et al., 2026) and limitations of standard guidance parameterization along student trajectories in on-policy distillation (Li et al., 2026a). Most closely related, prompt-aware adaptive CFG (Zhang & Li, 2025) varies guidance by conditioning input, like CAGS's per-class adaptation, but targets generation quality rather than dataset distillation. Sergievskii et al. (2026) systematically compared CFG variants in latent diffusion models. These insights remain unexplored in dataset distillation, where MGD^3 fixes λ across classes and timesteps. CAGS introduces a per-class complexity metric targeting discriminative feature preservation rather than aesthetic fidelity.

3 Method

AGS-DD replaces the globally fixed classifier-free guidance of training-free MGD^3 (Chan-Santiago et al., 2025) with complexity-based per-class guidance via CAGS (Section 3.2). Exploratory TAGS and IAST (Section 3.3) address temporal and IPC-budget adaptation, respectively, but are not recommended after negative ablations. See Figure 1 for the framework and Appendix C for the full sampling algorithm.

3.1 Background: Diffusion Models and Classifier-Free Guidance

Denoising diffusion probabilistic models (Ho et al., 2020) learn to reverse gradual noising with a neural network $\epsilon_\theta(\mathbf{x}_t, t, c)$ predicting noise at each timestep conditioned on class c. Classifier-free guidance (CFG) (Ho & Salimans, 2022) interpolates conditional and unconditional score predictions to enhance conditional generation:

$$\tilde{\epsilon}_\theta(\mathbf{x}_t, t, c) = (1 + \lambda)\,\epsilon_\theta(\mathbf{x}_t, t, c) - \lambda\,\epsilon_\theta(\mathbf{x}_t, t, \varnothing), \tag{1}$$

where $\lambda \geq 0$ is the guidance scale and $\varnothing$ the null class. Increasing λ trades diversity for conditioning fidelity. MGD^3 (Chan-Santiago et al., 2025) samples from a frozen pretrained DiT-XL/2 (Peebles & Xie, 2023) with fixed λ across classes and timesteps, starting at a high-noise timestep t_{start}.

3.2 CLASS-ADAPTIVE GUIDANCE STRENGTH (CAGS)

Classes differ in structural complexity, so no single guidance scale serves all optimally. Stronger guidance sharpens discriminative features in classes with high intra-class variance (e.g., morphologically diverse "dog breeds"), but may over-sharpen classes with low variance and high inter-class separability (e.g., "parachute") under the same guidance level.

Multi-factor complexity metric. We compute four complexity factors for each class c from real training images using VAE feature extraction and K-means clustering with silhouette-selected $K \in [2, 20]$: **mode count** (κ_c, number of visual modes), **cluster entropy** (H_c, balance of cluster sizes), **intra-class variance** (v_c, feature dispersion), and **inter-class separability** (d_c, distance to nearest class centroid normalized by intra-class spread). Systematic factor ablation (Section 5.3) and weight grid search (Appendix F) identify only entropy and intra-class variance as effective: mode count is nearly constant (K=2 for 94% of classes), and separability yields near-constant guidance (std 0.006). The optimal configuration $(\alpha, \beta, \gamma, \delta) = (0, 0.5, 0.5, 0)$ equally weights entropy and variance, yielding:

$$\rho_c = \sigma\Big(s \cdot \Big(0.5\,\hat{H}_c + 0.5\,\hat{v}_c\Big) - c_0\Big), \tag{2}$$

where $\hat{H}_c$ and $\hat{v}_c$ are min-max normalized, $\sigma(\cdot)$ is the sigmoid, $s = 3.0$ the slope, and $c_0 = 0.6$ the center. Guidance is $\lambda_c = \lambda_{\min} + (\lambda_{\max} - \lambda_{\min}) \cdot \rho_c$, with $[\lambda_{\min}, \lambda_{\max}] = [0.0, 0.06]$. The sigmoid provides a bounded, monotonic approximation to the theoretical linear scaling $\lambda_c^* \propto v_c/\sigma_c^2$ (Proposition 2), mapping unbounded complexity scores to valid guidance; see Appendix B.7 for the theory-to-implementation discussion. Classes with higher complexity receive stronger guidance. The general four-factor formulation is $\rho_c = \sigma\Big(s \cdot (\alpha\hat{\kappa}_c + \beta\hat{H}_c + \gamma\hat{v}_c - \delta\hat{d}_c) - c_0\Big)$; separability is subtracted because greater separability implies lower complexity.

Theoretical motivation. Section 4 proves that $\lambda_c^* \propto v_c/\sigma_c^2$ is increasing in both v_c and H_c (Proposition 2), and the adaptivity gap grows with cross-class heterogeneity (Theorem 1).

3.3 ADDITIONAL EXPLORATIONS: TAGS AND IAST

Both exploratory modules are *not recommended* after negative ablations (Section D.2). TAGS (Timestep-Adaptive Guidance Schedule) replaces constant λ with a time-varying $\lambda(t)$, concentrating guidance at high-noise timesteps. None of the three evaluated schedules improves over CAGS-alone: optimal λ_c^* depends on class properties, not the noise level. IAST (IPC-Adaptive Guidance Range) scales the guidance range by a logarithmic factor of IPC; at IPC=1 ($s \approx 0.29$), it eliminates 92% of per-class differentiation and hurts performance.

3.4 CAGS SAMPLER

Algorithm 1 (Appendix) integrates CAGS into a single sampling procedure using a frozen pretrained DiT, without gradients or weight updates. The only per-dataset computation is a one-time, cached complexity analysis. CAGS thus preserves MGD^3's training-free deployment simplicity while substantially improving distilled dataset quality through per-class guidance.

4 THEORETICAL ANALYSIS

We develop a formal framework explaining why per-class adaptive guidance outperforms fixed guidance, why cluster entropy and intra-class variance are the effective complexity factors, and why inter-class separability is not. Our analysis proceeds under a Gaussian mixture model for the class-conditional distribution (Ho & Salimans, 2022; Peebles & Xie, 2023).

Setup. Consider a class c whose conditional distribution is a K_c-component Gaussian mixture:

$$p(\mathbf{x} \mid c) = \sum_{k=1}^{K_c} \pi_{c,k}\, \mathcal{N}(\mathbf{x};\, \boldsymbol{\mu}_{c,k},\, \boldsymbol{\Sigma}_{c,k}), \tag{3}$$

where $\pi_{c,k} > 0$ are mode proportions satisfying $\sum_k \pi_{c,k} = 1$. The cluster entropy $H_c = -\sum_k \pi_{c,k} \log \pi_{c,k}$ measures the balance of mode sizes, and the intra-class variance $v_c = \mathrm{tr}[\bar{\boldsymbol{\Sigma}}_c]$ measures the total feature spread. Under CFG with scale λ, the effective conditional distribution is $p_\lambda(\mathbf{x} \mid c) \propto p(\mathbf{x} \mid c)^{1+\lambda}\, p(\mathbf{x})^{-\lambda}$.

Lemma 1 (CFG Mode Sharpening and Reweighting)**.** *For well-separated modes, the CFG-modified distribution is approximately* $p_\lambda(\mathbf{x} \mid c) \approx \sum_k \tilde{\pi}_{c,k}(\lambda)\, \mathcal{N}\Big(\mathbf{x};\, \boldsymbol{\mu}_{c,k},\, \frac{\boldsymbol{\Sigma}_{c,k}}{1+\lambda}\Big)$*, where* $\tilde{\pi}_{c,k}(\lambda) = \pi_{c,k}^{1+\lambda} \big/ \sum_j \pi_{c,j}^{1+\lambda}$*. Thus CFG has two effects: (i)* sharpening*—each mode's covariance is scaled by* $1/(1+\lambda)$*—and (ii)* reweighting*—dominant modes receive proportionally more weight.*

Proposition 1 (Diversity Cost of Guidance)**.** *The KL divergence* $D_c(\lambda) := \mathrm{KL}(\boldsymbol{\pi}_c \,\|\, \tilde{\boldsymbol{\pi}}_c(\lambda))$ *satisfies* $D_c(\lambda) = \lambda\, H_c + \log Z_c(\lambda)$*, with* $D_c''(0) = \mathrm{Var}_{\boldsymbol{\pi}_c}[\log \pi_{c,k}]$*, which is* decreasing in H_c*: balanced (high-entropy) classes incur lower diversity cost.*

Proposition 2 (Optimal Per-Class Guidance)**.** *Define the per-class generation utility* $U_c(\lambda) = (1+\lambda)\, v_c - \beta\, D_c(\lambda)$*, where the first term captures the* discriminative benefit *of guidance—sharper modes (covariance scaled by* $1/(1+\lambda)$ *per Lemma 1) increase the effective feature separation by a factor of* $(1+\lambda)$*, measured by the trace of the sharpened covariance* v_c*—and the second term captures the* diversity cost *from mode reweighting (Proposition 1). The linear-in-*λ *form of the benefit term is motivated by the observation that, for small* λ*, the covariance sharpening* $1/(1+\lambda) \approx 1-\lambda$ *produces a first-order reduction in mode spread proportional to* $\lambda \cdot v_c$*, so the discriminative benefit is approximately linear in* λ *in this regime. The parameter* $\beta > 0$ *controls the trade-off between discriminative sharpness and distributional fidelity. The optimal guidance* $\lambda_c^* = \arg\max_\lambda U_c(\lambda)$ *satisfies, to second order:*

$$\lambda_c^* \approx \frac{v_c}{\beta\, \sigma_c^2}, \tag{4}$$

where $\sigma_c^2 = \mathrm{Var}_{\boldsymbol{\pi}_c}[\log \pi_{c,k}]$*. Consequently,* λ_c^* *is* increasing in v_c *(intra-class variance) and* increasing in H_c *(cluster entropy, since* σ_c^2 *decreases with* H_c*). The linear-in-*λ *benefit term is valid for small* λ *(the regime used in practice,* $\lambda \in [0, 0.06]$*); for large* λ*, higher-order terms in* $D_c(\lambda)$ *dominate and the utility eventually decreases, consistent with the empirical finding that excessive guidance hurts performance (Fixed* $\lambda = 0.10$ *underperforms unguided, Table 1).*

Theorem 1 (Adaptivity Gap)**.** *Assume* $\sigma_c^2 > 0$*. Let* $\bar{\lambda} = \frac{\sum_{c=1}^C \sigma_c^2 \lambda_c^*}{\sum_{c=1}^C \sigma_c^2}$ *be the optimal shared guidance under the second-order approximation. The* adaptivity gap*—the average utility gain from per-class adaptation—is* $\Delta = \frac{1}{C}\sum_{c=1}^C \frac{\beta \sigma_c^2}{2}(\lambda_c^* - \bar{\lambda})^2 \geq 0$*, which equals zero iff all classes have identical optimal guidance and grows with the* σ_c^2*-weighted dispersion of* λ_c^**.*

Lemma 2 (Separability Invariance)**.** *Under the well-separated modes assumption, the optimal per-class guidance* λ_c^* *depends only on intra-class properties (*v_c*,* H_c*) and is independent of inter-class separability* d_c*.*

Connection to empirical findings. These results explain five key observations: *(1)* cluster entropy is the most effective single factor because high-entropy classes have near-zero diversity cost and sustain stronger guidance (Proposition 1); *(2)* separability is the least effective factor because λ_c^* is independent of it (Lemma 2); *(3)* the two-factor design (H_c, v_c) is optimal since λ_c^* is increasing in both (Proposition 2); *(4)* Across our ImageNet subsets, CAGS gains increase with cross-class heterogeneity rather than class count per se (Theorem 1); and *(5)* CAGS is neutral on the 10-class ImageNet-100 subset where complexity variation is narrow ($\Delta \approx 0$), while datasets with the same class count but broader complexity variation (e.g., ImageNette, $+5.28\%$) show substantial gains, confirming that the adaptivity gap depends on cross-class heterogeneity rather than class count alone. We validate the adaptivity gap prediction by measuring mode structure heterogeneity across datasets (Appendix B): the fraction of classes with multi-mode structure ($K > 2$) increases from 0% at $C{=}10$ to 60% at $C{=}1000$, tracking the actual CAGS gain.

Why TAGS and IAST fail. TAGS varies λ across timesteps but applies the same $\lambda(t)$ to all classes. Since λ_c^* depends on intra-class properties, not the noise level, the optimal $\lambda(t)$ at each timestep is the cross-class average $\bar{\lambda}$, yielding no improvement over fixed guidance. IAST compresses the guidance range by a factor $s(\mathrm{IPC})$, reducing the adaptivity gap to $s^2 \cdot \Delta_{\mathrm{CAGS}}$; at IPC=1, $s \approx 0.29$, eliminating 92% of the per-class differentiation. Both modules vary λ along dimensions where the optimal λ_c^* is constant. Proofs are in Appendix B.

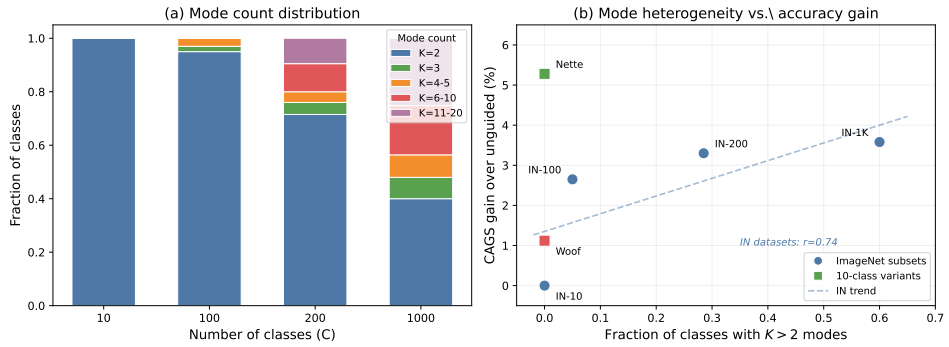

Figure 2: Quantitative validation of Theorem 1. (a) Distribution of mode counts (K) across ImageNet subsets with increasing class count. (b) Mode heterogeneity (fraction of classes with $K > 2$) vs. actual CAGS accuracy gain.

5 Experiments

5.1 Experimental Setup

Datasets. We evaluate on ImageNet-100 (10-class and 100-class subsets) (Russakovsky et al., 2015), ImageNette, ImageWoof, a 200-class ImageNet subset, and full ImageNet-1K (1000 classes). Beyond ImageNet, we evaluate on Food-101 (Bossard et al., 2014) (101 food categories) and CIFAR-10 (Krizhevsky, 2009) (10 object categories) using Stable Diffusion (Rombach et al., 2022) 1.5. All images are resized to 256×256 pixels.

Architecture and protocol. Following MGD^3 (Chan-Santiago et al., 2025), our frozen generator for ImageNet-derived experiments is pretrained DiT-XL/2 (Peebles & Xie, 2023) (50 denoising steps; high-noise guidance over the first 25 timesteps). ConvNet-6, ResNet-18, and ResNetAP-10 (He et al., 2016) use instance normalization and SGD (lr 0.1, weight decay $1e{-4}$, batch size 128) with a multi-step LR schedule. Training lasts 2000 epochs for 100-class experiments, ImageNette, and ImageWoof; 3000 for the 10-class subset and CIFAR-10; 1000 for IN-1K. Augmentations include random resized crop, horizontal flip, color jitter, PCA lighting noise, and CutMix. Following MGD^3, we report *best*-epoch top-1 accuracy as mean $\pm$ standard deviation: five random seeds for main 100-class IPC=10 experiments, three otherwise (population std for reporting; sample std for t-tests).

CAGS configurations. CAGS uses complexity weights $(\alpha, \beta, \gamma, \delta) = (0, 0.5, 0.5, 0)$, optimal in factor ablation (Section 5.3) and weight grid search (Appendix F), with sigmoid parameters $s = 3.0$, $c_0 = 0.6$, and guidance range $[0.0, 0.06]$.

5.2 Main Results: ImageNet-100 (100-class)

Table 1 compares ImageNet-100 (100-class, IPC=10) results with baselines published in the MGD^3 paper (Chan-Santiago et al., 2025). Fixed $\lambda = 0.10$ yields $21.31 \pm 0.42\%$ on ConvNet-6, underperforming unguided generation and showing that globally fixed guidance is suboptimal. CAGS achieves $25.55 \pm 0.21\%$, $30.81 \pm 0.43\%$, and $29.74 \pm 0.46\%$ on ConvNet-6, ResNet-18, and ResNetAP-10, respectively, surpassing all published baselines. On ResNet-18, 30.81% vs. MGD^3's 23.6% gives $+7.21\%$ absolute improvement. Greater gains over unguided on deeper architectures (ResNet-18: $+16.2\%$ relative vs. ConvNet-6: $+11.6\%$) indicate that CAGS-generated images contain richer discriminative features better exploited by expressive classifiers. On the 10-class ImageNet-100 subset, CAGS is neutral (-0.7% relative at IPC=10): limited class complexity variation leaves no benefit from per-class adaptation (Appendix D.1).

5.3 Complexity Factor Ablation

Table 2 compares single-factor, equal-weight, and variance-dominated configurations (ImageNet-100, 100-class, IPC=10, ConvNet-6, 3 seeds). All outperform unguided (22.79%, 3 seeds), supporting per-

Table 1: ImageNet-100 results (100-class, IPC=10). Published baseline results are taken from Table 2 of Chan-Santiago et al. (2025); D^4M: our same-protocol implementation. Seeds: 5 for CAGS and Unguided; 3 for D^4M, Fixed guidance, and hybrid.

Method	**ConvNet-6**	**ResNet-18**	**ResNetAP-10**
Published baselines (from MGD^3 paper)			
Random	17.0 ± 0.3	17.5 ± 0.5	19.1 ± 0.4
Herding	17.2 ± 0.3	16.1 ± 0.2	19.8 ± 0.3
IDC-1 (Kim et al., 2022)	24.3 ± 0.5	25.1 ± 0.2	25.7 ± 0.1
MinMaxDiff (Gu et al., 2024)	22.3 ± 0.5	22.5 ± 0.3	24.8 ± 0.2
MGD^3 (Chan-Santiago et al., 2025)	23.4 ± 0.9	23.6 ± 0.4	25.8 ± 0.5
Same protocol (same DiT generator, 2000 epochs)			
D^4M (Su et al., 2024) (t=40)	23.07 ± 0.59	—	—
Unguided	22.90 ± 0.50	26.52 ± 0.27	26.51 ± 0.43
Fixed $\lambda = 0.10$	21.31 ± 0.42	—	—
CAGS	$\mathbf{25.55 \pm 0.21}$	$\mathbf{30.81 \pm 0.43}$	$\mathbf{29.74 \pm 0.46}$
CAGS+D^4M hybrid (t=45)	25.17 ± 0.37	30.18 ± 0.17	29.51 ± 0.22

Table 2: Complexity factor ablation (100-class, IPC=10, ConvNet-6, 3 seeds). ρ std is the standard deviation of per-class complexity ρ_c, before mapping to $\lambda_c = 0.06\,\rho_c$. Independent datasets with different diffusion sampling seeds yield $\sim 1\%$ differences for "Full CAGS" and "Entropy only" relative to the weight grid search (Appendix F). Independent dataset generation also explains the unguided difference: 22.79% (3 seeds) vs. 22.90% (5 seeds, Table 1).

Configuration	α	β	γ	δ	**Top-1 (%)**	ρ **std**
Unguided	—	—	—	—	22.79 ± 0.18	—
Entropy only	0	1	0	0	$\mathbf{25.31 \pm 0.62}$	0.042
Equal weights	0.25	0.25	0.25	0.25	24.69 ± 0.29	0.043
No separability	0.25	0.25	0.50	0	24.61 ± 0.44	0.081
Intra-var only	0	0	1	0	24.46 ± 0.16	0.154
Sep-dominated	0.10	0.10	0.20	0.60	24.25 ± 0.44	0.034
Full CAGS	0.30	0.30	0.20	0.20	24.24 ± 0.26	0.035
Mode count only	1	0	0	0	23.92 ± 0.12	0.011
Separability only	0	0	0	1	23.75 ± 0.09	0.006

class adaptive guidance. Cluster entropy alone has the highest mean accuracy ($25.31 \pm 0.62\%$) and seed variance; inter-class separability alone is weakest ($23.75 \pm 0.09\%$), with near-constant per-class guidance (std 0.006), effectively fixed at $\lambda \approx 0.04$. All three configurations reducing or eliminating separability weight outperform the full four-factor design (24.24%); separability-dominated matches it. Separability is thus least effective: per-class guidance variation, rather than factor count, drives improvement.

The optimal two-factor design $(0, 0.5, 0.5, 0)$ weights entropy and intra-class variance, the factors with the widest per-class spread, maximizing the guidance range. Per-class λ variation is $2.3\times$ larger (std 0.0049 vs. 0.0021 for four factors, from cached complexity scores), yielding larger accuracy gains. Class-level analysis (Figure 4, Appendix D.6) confirms stronger guidance and larger gains for higher-complexity quartiles ($2.8\times$ between top and bottom halves).

5.4 Statistical Significance

With five seeds for main 100-class IPC=10 experiments and three otherwise, 100-class gains are significant at all IPC settings ($p < 0.05$) with large effects (Cohen's $d \geq 3.90$). All three architectures show highly significant gains with 5 seeds ($p < 0.001$, Cohen's $d \geq 6.2$); the neutral 10-class IPC=10 result is non-significant ($p = 0.721$).

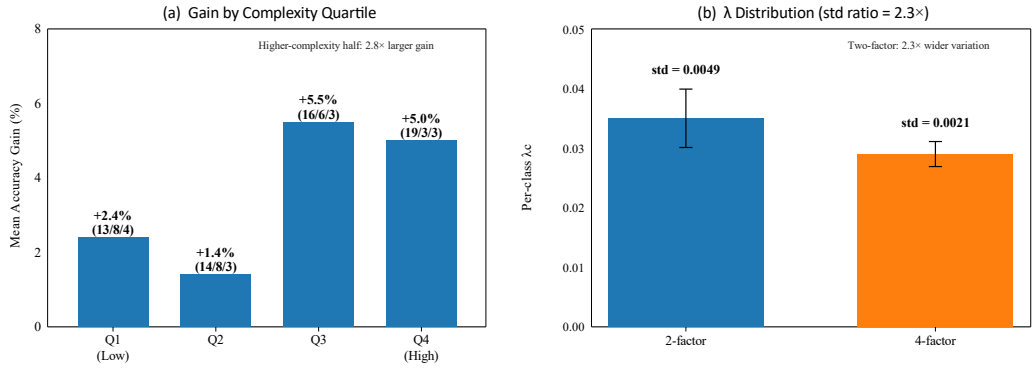

Figure 3: Class-level mechanism analysis. (a) Mean accuracy gains by complexity quartile: higher-complexity Q3–Q4 benefit $2.8\times$ more than Q1–Q2. (b) Per-class guidance λ_c: two factors yield $2.3\times$ larger variation than four (std 0.0049 vs. 0.0021).

Table 3: Paired two-tailed t-tests: CAGS vs. unguided and IAST+CAGS vs. CAGS. Seeds: 5 for the three cross-architecture entries (100-cls IPC=10, ResNet-18, ResNetAP-10); 3 otherwise. Guidance range: $[0.0, 0.08]$ for 10-cls (matching 100-class average guidance); $[0.0, 0.06]$ otherwise.

Setting	**Comparison**	**Δ (%)**	**t-stat**	**p-value**	**Cohen's d**
100-cls IPC=1 (ConvNet-6)	CAGS vs Ung	$+0.72$	6.80	0.021	3.93
100-cls IPC=10 (ConvNet-6)	CAGS vs Ung	$+2.65$	13.97	1.5×10^{-4}	6.21
100-cls IPC=50 (ConvNet-6)	CAGS vs Ung	$+6.74$	21.01	0.002	20.51
100-cls IPC=1 (ConvNet-6)	IAST+CAGS vs CAGS	-0.70	-4.17	0.053	-2.41
ResNet-18 IPC=10	CAGS vs Ung	$+4.29$	19.79	3.9×10^{-5}	10.63
ResNetAP-10 IPC=10	CAGS vs Ung	$+3.23$	16.35	8.2×10^{-5}	6.46
10-cls IPC=1	CAGS vs Ung	-2.93	-4.21	0.052	-2.43
10-cls IPC=10	CAGS vs Ung	-0.33	-0.41	0.721	-0.24
10-cls IPC=50	CAGS vs Ung	$+5.13$	3.42	0.076	1.97
ImageNette	CAGS vs Ung	$+5.28$	12.33	0.007	8.58
ImageWoof	CAGS vs Ung	$+1.12$	4.50	0.046	4.33

5.5 D^{4}M Comparison and 2026 Methods

Under our exact protocol, D^{4}M (Su et al., 2024) peaks at t_{start}=40 with 23.07%, below CAGS and barely above unguided (22.90%). Appendix E details D^{4}M sensitivity, cross-dataset comparisons, and comparisons with 2026 methods (DMGD (Wang et al., 2026), LGD (Chan-Santiago & Shah, 2026)). DMGD's higher absolute ImageNette/ImageWoof accuracy reflects a 5.1 percentage-point higher baseline due to lighter augmentation; CAGS's relative improvement ($+14.8\%$) is comparable to DMGD's ($+15.7\%$) while adding negligible overhead.

5.6 CAGS+D^{4}M Hybrid Ablation

To test D^{4}M's real-image initialization with CAGS, we use VAE encode $\rightarrow$ add noise at t_{start} $\rightarrow$ denoise with per-class CAGS guidance, sweeping $t_{\text{start}} \in \{35, 40, 45\}$ on ConvNet-6. The optimum, t_{start}=45 ($25.17 \pm 0.37\%$), is evaluated across all three architectures (Table 1). Gains over CAGS are negative: -0.38%, -0.63%, and -0.23% on ConvNet-6, ResNet-18, and ResNetAP-10, respectively, with $2.75\times$ higher variance on the latter. CAGS's training-free, pure-noise approach is thus simpler and more effective.

Table 4: Scaling of CAGS. IPC=10, instance normalization. Epochs: 2000 for $\leq$200 classes, 1000 for 1000 classes. Seeds: 5 for IN-100; 3 for ImageNette/ImageWoof/IN-200/IN-1K (ResNet-18, ResNetAP-10); 2 for IN-1K ConvNet-6. ImageNette/ImageWoof (10 classes) have broader complexity variation than the 10-class ImageNet-100 subset (Theorem 1).

Dataset	**Classes**	**Unguided**	**CAGS**	Δ
ImageNette	10	49.99 ± 0.22	$\mathbf{55.27 \pm 0.68}$	$+5.28$
ImageWoof	10	31.75 ± 0.06	$\mathbf{32.87 \pm 0.29}$	$+1.12$
IN-100	100	22.90 ± 0.50	$\mathbf{25.55 \pm 0.21}$	$+2.65$
IN-200	200	18.49 ± 0.21	$\mathbf{21.79 \pm 0.16}$	$+3.30$
ImageNet-1K (1000 classes), 1000 epochs				
IN-1K (ConvNet-6)	1000	11.48 ± 0.16	$\mathbf{15.06 \pm 0.06}$	$\mathbf{+3.58}$
IN-1K (ResNet-18)	1000	16.86 ± 0.06	$\mathbf{23.83 \pm 0.13}$	$\mathbf{+6.97}$
IN-1K (ResNetAP-10)	1000	15.64 ± 0.08	$\mathbf{22.46 \pm 0.18}$	$\mathbf{+6.83}$

5.7 Scaling Analysis: From 10 to 1000 Classes

Table 4 reports CAGS results from 10 to 1000 classes with otherwise fixed settings (IPC=10, instance normalization; IN-1K: 1000 epochs). Gains grow from $+2.65\%$ (100 classes) to $+3.58\%$ (1000 classes, ConvNet-6), rising on deeper architectures to $+6.97\%$ ($+41.3\%$ relative) on ResNet-18 and $+6.83\%$ ($+43.7\%$) on ResNetAP-10, consistent with Theorem 1. The 10-class ImageNette ($+5.28\%$) and ImageWoof ($+1.12\%$) deviate because the adaptivity gap depends on *cross-class heterogeneity* in λ_c^*, not class count alone. The 10-class ImageNet-100 subset (Appendix D.1) has narrow complexity variation (0% gain); ImageNette's broader intra-class variance heterogeneity yields gains exceeding the 100-class subset. Food-101 and CIFAR-10 results with Stable Diffusion 1.5 confirm this pattern beyond ImageNet (Table 13, Appendix D.9).

6 Limitations

Our two-factor default is $(\alpha{=}0, \beta{=}0.5, \gamma{=}0.5, \delta{=}0)$; four factors may suit high mode-count or separability variation (Appendix H). Main 100-class results use five seeds; some supplementary experiments use three. As in MGD^3, best-epoch top-1 reporting aids comparability but may overestimate performance versus final-epoch evaluation. In a main 100-class ConvNet-6 spot-check (seed 0), CAGS vs. unguided yields 24.92% vs. 22.86% (final epoch, $+2.06\%$) and 25.22% vs. 23.62% (best epoch): inflation is 0.30% vs. 0.76%, preserving the ranking. Theory assumes well-separated Gaussian modes and a second-order approximation, with linear-in-λ benefits restricted to small λ ($\lambda \in [0, 0.06]$). Non-ImageNet results use SD 1.5; low Food-101 accuracy (4.17%) reflects the generator's domain gap rather than a CAGS limitation. The 10-class boundary reflects cross-class complexity variation (Theorem 1), not class count: ImageNette (10 classes, $+5.28\%$) benefits; the 10-class ImageNet-100 subset gains 0% with narrow variation. Its IPC=1 decline (-2.93%, $p{=}0.052$) suggests IPC-dependent benefits: adaptation may overfit guidance to noisy complexity estimates at very low budgets. Section 5.1 lists hyperparameters.

7 Conclusion

CAGS replaces MGD^3's globally fixed classifier-free guidance with per-class guidance based on cluster entropy and intra-class variance, identified as optimal by four-factor ablation. Theoretically, $\lambda_c^* \propto v_c/\sigma_c^2$ increases with both factors, and the adaptivity gap grows with cross-class heterogeneity. CAGS surpasses all published ImageNet-100 baselines on three architectures (30.81% on ResNet-18, 29.74% on ResNetAP-10, 25.55% on ConvNet-6), reaching $+31$–$+44\%$ relative gains at 1000 classes. Benefits depend on complexity variation, not class count: neutral on 10-class ImageNet-100 but $+5.28\%$ on ImageNette. CAGS retains MGD^3's training-free, single-pass operation with negligible overhead and orthogonality to distribution matching and sample selection. Future work includes automatic Bayesian weight optimization, prompt-level adaptive text-to-image guidance, and combining CAGS with distribution matching to jointly optimize guidance and latent alignment.

REPRODUCIBILITY STATEMENT

All hyperparameters, random seeds, hardware specifications, and data processing steps are documented in Section 5.1 and Appendix A. The complete sampling algorithm is in Appendix C. An anonymous code entry is provided in the abstract. AI assistance is disclosed in the AI Use Statement before the references.

AI USE STATEMENT

In this work, we used generative AI tools (LLMs) for: (1) implementing and debugging the CAGS complexity analysis pipeline, dataset generation scripts, and classifier training code; (2) formatting LaTeX proofs and mathematical derivations; (3) drafting and editing portions of the manuscript for clarity; and (4) formatting references. We did not use AI tools for generating experimental results, fabricating data, or drawing scientific conclusions. All experimental design, data analysis, theoretical development, and interpretation of results were conducted solely by the authors. We have verified the correctness of all AI-assisted code, proofs, and text, and take responsibility for the final content.

REFERENCES

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pp. 446–461. Springer, 2014.

George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A Efros, and Jun-Yan Zhu. Dataset distillation by matching training trajectories. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10718–10727. IEEE, 2022.

Jeffrey A Chan-Santiago and Mubarak Shah. Learnability-guided diffusion for dataset distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 41657–41666, 2026.

Jeffrey A Chan-Santiago, Praveen Tirupattur, Gaurav Kumar Nayak, Gaowen Liu, and Mubarak Shah. MGD^3: Mode-guided dataset distillation using diffusion models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 52861–52876. PMLR, 2025. URL `https://proceedings.mlr.press/v267/santiago25a.html`.

Nicolas Dufour, Alexei A Efros, and Patrick Pérez. The fid lottery: Quantifying hidden randomness in generative-model evaluation. *arXiv preprint arXiv:2606.20536*, 2026.

Junqiao Fan, Yunjiao Zhou, Min Chang Jordan Ren, and Jianfei Yang. Generative dataset distillation using min-max diffusion model. In *European Conference on Computer Vision*, pp. 60–67. Springer, 2024.

Jianyang Gu, Saeed Vahidian, Vyacheslav Kungurtsev, Haonan Wang, Wei Jiang, Yang You, and Yiran Chen. Efficient dataset distillation via minimax diffusion. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15793–15803. IEEE, 2024.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Jang-Hyun Kim, Jinuk Kim, Seong Joon Oh, Sangdoo Yun, Hwanjun Song, Joonhyun Jeong, Jung-Woo Ha, and Hyun Oh Song. Dataset condensation via efficient synthetic-data parameterization. In *International Conference on Machine Learning*, pp. 11102–11118. PMLR, 2022.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Bingnan Li, Haozhe Wang, Haozhong Xiong, Fangtai Wu, Jinpeng Yu, Yang Shi, Jiaming Liu, and Ruihua Huang. Rethinking classifier-free guidance in on-policy diffusion distillation. *arXiv preprint arXiv:2607.24731*, 2026a.

Mingzhuo Li, Guang Li, Linfeng Ye, Jiafeng Mao, Takahiro Ogawa, Konstantinos N Plataniotis, and Miki Haseyama. Sas: Semantic-aware sampling for generative dataset distillation. *IEEE Open Journal of Signal Processing*, 2026b.

Pengxiang Li, Shilin Yan, Jiayin Cai, Renrui Zhang, Ruichuan An, Ziyu Guo, and Xiaowei Gao. Adaptive classifier-free guidance via dynamic low-confidence masking. *Advances in Neural Information Processing Systems*, 38:52874–52893, 2025a.

Wenmin Li, Shunsuke Sakai, Zhongkai Zhao, and Tatsuhito Hasegawa. Pool-select-refine for allocation-aware generative dataset distillation. *arXiv preprint arXiv:2606.01920*, 2026c.

Zekai Li, Xinhao Zhong, Samir Khaki, Zhiyuan Liang, Yuhao Zhou, Mingjia Shi, Ziqiao Wang, Xuanlei Zhao, Wangbo Zhao, Ziheng Qin, et al. Dd-ranking: Rethinking the evaluation of dataset distillation. *arXiv preprint arXiv:2505.13300*, 2025b.

Yanqing Liu, Jianyang Gu, Kai Wang, Zheng Zhu, Wei Jiang, and Yang You. Dream: Efficient dataset distillation by representative matching. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17314–17324. IEEE, 2023.

Dawid Malarz, Artur Kasymov, Maciej Zięba, Jacek Tabor, and Przemysław Spurek. Classifier-free guidance with adaptive scaling. In *ECAI 2025*, pp. 435–442, 2025.

Brian B Moser, Shalini Sarode, Federico Raue, Stanislav Frolov, Krzysztof Adamkiewicz, Arundhati Shanbhag, Joachim Folz, Tobias C Nauen, and Andreas Dengel. PRISM: Diversifying dataset distillation by decoupling architectural priors. *Transactions on Machine Learning Research*, 2026.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205. IEEE, 2023.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10684–10695. ieee, 2022.

Ayush Roy, Wei-Yang Alex Lee, Rudrasis Chakraborty, and Vishnu Suresh Lokhande. Manifoldgd: Training-free hierarchical manifold guidance for diffusion-based dataset distillation. *arXiv preprint arXiv:2602.23295*, 2026.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

Artem Sergievskii, Artyom Turevich, and Sergey Kastryulin. Revisiting classifier-free guidance methods in latent diffusion models. *arXiv preprint arXiv:2608.16786*, 2026.

Abdullah Al Shafi, Kazi Saeed Alam, Sk Imran Hossain, and Engelbert Mephu Nguifo. Dash: Dual-branch score distillation for guidance-calibrated compact diffusion models. *arXiv preprint arXiv:2606.00798*, 2026.

Duo Su, Junjie Hou, Weizhi Gao, Yingjie Tian, and Bowen Tang. D^4M: Dataset distillation via disentangled diffusion model. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5809–5818. IEEE, 2024.

Qichao Wang, Yunhong Lu, Hengyuan Cao, Junyi Zhang, and Min Zhang. DMGD: Train-free dataset distillation with semantic-distribution matching in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12417–12427, 2026.

Max Welling. Herding dynamic weights for partially observed random field models. *arXiv preprint arXiv:1205.2605*, 2012.

Shai Yehezkel, Omer Dahary, Andrey Voynov, and Daniel Cohen-Or. Navigating with annealing guidance scale in diffusion space. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pp. 1–11, 2025.

Zeyuan Yin, Eric Xing, and Zhiqiang Shen. Squeeze, recover and relabel: Dataset condensation at imagenet scale from a new perspective. *Advances in Neural Information Processing Systems*, 36: 73582–73603, 2023.

Huixuan Zhang and Xiaojun Wan. How much to guide: Revisiting adaptive guidance in classifier-free guidance text-to-vision diffusion models. In *Proceedings of the 7th ACM International Conference on Multimedia in Asia*, pp. 1–8, 2025.

Xuanhao Zhang and Chang Li. Prompt-aware classifier free guidance for diffusion models. *arXiv preprint arXiv:2509.22728*, 2025.

Ganlong Zhao, Guanbin Li, Yipeng Qin, and Yizhou Yu. Improved distribution matching for dataset condensation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7856–7865. IEEE, 2023.

Letian Zhou, Songhua Liu, and Xinchao Wang. Coda: From text-to-image diffusion models to training-free dataset distillation. In *International Conference on Learning Representations*, volume 2026, pp. 81947–81980, 2026.

A Reproducibility Details

Models and data. All experiments use a frozen pretrained DiT-XL/2 (Peebles & Xie, 2023) for ImageNet-derived experiments and Stable Diffusion 1.5 (Rombach et al., 2022) for non-ImageNet experiments, both publicly available. The complexity analysis pipeline (VAE feature extraction, K-means clustering, sigmoid mapping) uses only standard libraries (PyTorch, scikit-learn).

Hyperparameters. We report all hyperparameters: sigmoid slope s=3.0, center c_0=0.6, guidance range $[0.0, 0.06]$, complexity weights (α=0, β=0.5, γ=0.5, δ=0), 50 denoising steps with a 25-step high-noise guidance window, and training with SGD (lr 0.1, weight decay 1e$-$4, batch size 128). Classifier training uses 2000 epochs for 100-class experiments, ImageNette, and ImageWoof; 3000 for 10-class and CIFAR-10; 1000 for IN-1K. Data augmentation includes random resized crop, horizontal flip, color jitter, PCA lighting noise, and CutMix.

Random seeds. The main 100-class IPC=10 experiments (CAGS and Unguided on all three architectures) use 5 random seeds (seeds 0–4). All other experiments use 3 random seeds (seeds 0, 1, 2) except where noted in table captions (IN-1K ConvNet-6 uses 2 seeds; some sensitivity analysis entries use 2 seeds). Population std is used for reported standard deviations; sample std (ddof=1) is used for t-tests.

Hardware. All experiments run on NVIDIA RTX 4090 GPUs. The one-time complexity analysis takes approximately 22 minutes; dataset generation takes $\sim$16 minutes per configuration for 100 classes (IPC=10).

LLM usage. Large language models were used for code implementation and debugging, proof formatting, and manuscript editing. All experimental design, analysis, and scientific conclusions were conducted solely by the authors. Full disclosure is provided in the AI Use Statement in the main text.

B Proofs and Detailed Derivations

B.1 Proof of Lemma 1

For well-separated modes, $p(\mathbf{x} \mid c)^{1+\lambda} \approx \sum_k \pi_{c,k}^{1+\lambda} \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_{c,k}, \boldsymbol{\Sigma}_{c,k})^{1+\lambda}$ in each mode's support. Since $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})^{1+\lambda} \propto \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}/(1+\lambda))$ (a Gaussian raised to a power remains Gaussian with rescaled covariance), the result follows. The $p(\mathbf{x})^{-\lambda}$ term contributes a slowly-varying factor that we absorb into the normalization; this is exact when $p(\mathbf{x})$ is uniform and approximate otherwise.

B.2 Proof of Proposition 1

Substituting $\tilde{\pi}_k = \pi_k^{1+\lambda}/Z$ into the KL divergence yields $D_c(\lambda) = \sum_k \pi_k \log \pi_k - \sum_k \pi_k[(1+\lambda)\log \pi_k - \log Z] = \lambda H_c + \log Z$. For (a), $Z(0) = 1$ and $Z'(0) = \sum_k \pi_k \log \pi_k = -H_c$, so $D_c'(0) = H_c + Z'(0)/Z(0) = 0$. For (b), $Z''(0) = \sum_k \pi_k (\log \pi_k)^2$, giving $D_c''(0) = Z''(0) - Z'(0)^2 = \sum_k \pi_k (\log \pi_k)^2 - H_c^2 = \mathrm{Var}_{\boldsymbol{\pi}_c}[\log \pi_k]$. For (c), $\mathrm{Var}[\log \pi_k] = \mathbb{E}[(\log \pi_k)^2] - (\mathbb{E}[\log \pi_k])^2 = \mathbb{E}[(\log \pi_k)^2] - H_c^2$. Since $\log$ is concave, $\mathbb{E}[(\log \pi_k)^2]$ is Schur-convex and H_c is Schur-concave; as the distribution becomes more balanced (H_c increases), both terms converge, driving the variance to zero at maximum entropy (all $\pi_k = 1/K_c$).

B.3 Proof of Proposition 2

By Proposition 1, $D_c(\lambda) \approx \frac{1}{2}\sigma_c^2 \lambda^2$ for small λ (using $D_c(0) = D_c'(0) = 0$ and $D_c''(0) = \sigma_c^2$). Substituting into the utility: $U_c(\lambda) \approx v_c + v_c\lambda - \frac{\beta}{2}\sigma_c^2 \lambda^2$. Setting $U_c'(\lambda) = v_c - \beta\, \sigma_c^2\, \lambda = 0$ yields $\lambda_c^* = v_c/(\beta\, \sigma_c^2)$. The monotonicity claims follow: $\partial \lambda_c^*/\partial v_c = 1/(\beta\, \sigma_c^2) > 0$, and $\partial \lambda_c^*/\partial H_c = -v_c/(\beta\, \sigma_c^4) \cdot \partial \sigma_c^2/\partial H_c > 0$ since $\partial \sigma_c^2/\partial H_c < 0$ by Proposition 1(c).

B.4 Proof of Theorem 1

Since $U_c(\lambda)$ is a concave quadratic in λ (to second order), $U_c(\lambda) = U_c(\lambda_c^*) - \frac{\beta\sigma_c^2}{2}(\lambda - \lambda_c^*)^2$. Averaging over classes and noting $U_c(\lambda_c^*) - U_c(\bar{\lambda}) = \frac{\beta\sigma_c^2}{2}(\bar{\lambda} - \lambda_c^*)^2$ yields the adaptivity gap. Non-negativity follows from the squared term; equality holds iff $\lambda_c^* = \bar{\lambda}$ for all c.

B.5 Proof of Lemma 2

By Lemma 1, the effective mode weights $\tilde{\pi}_{c,k}(\lambda) = \pi_{c,k}^{1+\lambda} / \sum_j \pi_{c,j}^{1+\lambda}$ depend solely on the intra-class mode proportions $\boldsymbol{\pi}_c$. The diversity cost $D_c(\lambda)$ (Proposition 1) is therefore a function of $\boldsymbol{\pi}_c$ alone. Since $\lambda_c^* \approx v_c/(\beta\,\sigma_c^2)$ (Proposition 2) involves only v_c (intra-class variance) and σ_c^2 (a function of $\boldsymbol{\pi}_c$), inter-class separability does not enter. The $p(\mathbf{x})^{-\lambda}$ term in CFG involves the marginal $p(\mathbf{x}) = \sum_{c'} p(c')\,p(\mathbf{x}|c')$, which depends on all classes; however, for well-separated classes, this term contributes a class-dependent constant that is absorbed into the normalization (Lemma 1), leaving the mode structure—and hence the optimal λ_c^*—unaffected by inter-class geometry.

B.6 Remark on Assumptions

The well-separated modes assumption (Lemma 1) is an idealization; in practice, modes may overlap and the $p(\mathbf{x})^{-\lambda}$ term introduces inter-class effects. However, the qualitative predictions—that optimal λ increases with v_c and H_c, that separability is irrelevant, and that the adaptivity gap grows with class heterogeneity—are robust to relaxation of this assumption, as confirmed by experiments across datasets, architectures, and scales (Sections 5–E).

B.7 From Theory to Implementation

Proposition 2 predicts $\lambda_c^* \propto v_c/\sigma_c^2$, which is linear in v_c and entropy-dependent through σ_c^2. The implementation uses a sigmoid mapping $\rho_c = \sigma(s \cdot (0.5\hat{H}_c + 0.5\hat{v}_c) - c_0)$ followed by $\lambda_c = 0.06\,\rho_c$, which is not a direct instantiation of the linear formula. The sigmoid serves two practical purposes: it bounds $\lambda_c \in [0, 0.06]$ to prevent over-guidance (addressing the small-λ regime where the theory is valid), and it provides a smooth, monotonic mapping from the complexity score to the guidance scale that is robust to normalization choices. The key theoretical predictions—that λ_c should be increasing in both H_c and v_c, and independent of separability—are preserved by the sigmoid mapping since it is monotonic in its argument. The linear prediction from the theory provides the *qualitative* design rationale (which factors to include, which to exclude), while the sigmoid provides the *quantitative* robustness needed for practical deployment.

B.8 Quantitative Validation of Theorem 1

We empirically validate the adaptivity gap prediction by measuring mode structure heterogeneity across datasets of varying class count (Figure 2). On ImageNet-derived subsets, the fraction of classes with multi-mode structure ($K > 2$) increases monotonically: 0% at C=10 (the 10-class ImageNet-100 subset, where complexity variation is narrow), 5% at C=100, 28.5% at C=200, and 60% at C=1000, tracking the increasing diversity of mode structures. The actual CAGS gain increases in parallel: 0% on the 10-class ImageNet-100 subset, $+2.65\%$ at C=100, $+3.30\%$ at C=200, $+3.58\%$ at C=1000. Notably, the 10-class datasets ImageNette ($+5.28\%$) and ImageWoof ($+1.12\%$) show varying gains despite having K=2 for all classes, indicating that intra-class variance heterogeneity alone can drive the adaptivity gap—the zero-gain prediction applies specifically to the 10-class ImageNet-100 subset, whose classes happen to have narrow complexity variation, not to all 10-class datasets.

Table 5: ImageNet-100 (10-class) results with ConvNet-6, IPC=10, 3 seeds. Best-accuracy tracking per MGD^3 protocol. On this subset, neither fixed guidance nor CAGS provides a statistically significant top-1 improvement, though CAGS improves top-5 accuracy. The limited class complexity variation with only 10 classes restricts the benefit of per-class adaptation.

Method	**Avg. λ**	**Top-1 (%)**	**Top-5 (%)**	**Δ top-1**
Unguided	0.000	49.73 ± 1.20	83.20 ± 0.65	—
Fixed $\lambda = 0.05$	0.050	49.00 ± 0.71	83.60 ± 0.98	-0.73
CAGS $[0.0, 0.08]$	$\sim$0.055	49.40 ± 0.28	$\mathbf{84.87 \pm 0.19}$	-0.33

C AGS-DD Framework and Sampling Algorithm

Algorithm 1 AGS-DD Sampling (CAGS core)

Require: Frozen DiT ϵ_θ, class c, complexity cache $\{\rho_{c'}\}_{c'=1}^{C}$
Ensure: Synthetic image $\mathbf{x}_0$
$\lambda_c \leftarrow \lambda_{\min} + (\lambda_{\max} - \lambda_{\min}) \cdot \rho_c$ ▷ CAGS per-class guidance
$\mathbf{x}_{t_{\text{start}}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = t_{\text{start}}, t_{\text{start}} - 1, \ldots, 1$ **do**
  // Optional: $\lambda_t \leftarrow \text{TAGS}(\lambda_c, t)$ ▷ TAGS (not recommended)
  $\tilde{\epsilon} \leftarrow (1 + \lambda_c)\, \epsilon_\theta(\mathbf{x}_t, t, c) - \lambda_c\, \epsilon_\theta(\mathbf{x}_t, t, \varnothing)$
  $\mathbf{x}_{t-1} \leftarrow \text{DenoiseStep}(\mathbf{x}_t, \tilde{\epsilon}, t)$
end for
return $\mathbf{x}_0$

D Extended Experimental Results

D.1 Main Results: ImageNet-100 (10-class)

Table 5 presents results on the 10-class ImageNet-100 subset with ConvNet-6 and instance normalization, using the MGD^3 evaluation protocol (best top-1 accuracy across all training epochs). The unguided baseline achieves $49.73 \pm 1.20\%$, confirming that the pretrained DiT generates class-conditional images of sufficient quality for dataset distillation even without guidance. Fixed guidance at $\lambda = 0.05$ yields $49.00 \pm 0.71\%$, a -0.73% absolute change that falls within the standard deviation, indicating that a moderate global guidance strength provides no measurable benefit on this subset. CAGS at the matching average guidance range $[0.0, 0.08]$ achieves $49.40 \pm 0.28\%$, which is statistically indistinguishable from unguided (-0.33% absolute, within one standard deviation). However, CAGS does improve top-5 accuracy: $84.87 \pm 0.19\%$ vs. unguided $83.20 \pm 0.65\%$ ($+1.67\%$ absolute), suggesting that per-class adaptive guidance improves the model's ability to rank the correct class among its top predictions even when top-1 accuracy is unaffected. The neutral top-1 result on 10-class IPC=10 contrasts sharply with the strong gains observed on the 100-class subset (Section 5.2), and we attribute this to the limited class complexity variation when only 10 classes are present—a hypothesis we validate in Table 6.

IPC ablation reveals boundary conditions. Table 6 presents the full IPC ablation on the 10-class subset. At IPC=50, CAGS achieves $69.87 \pm 1.43\%$ vs. unguided $64.73 \pm 0.77\%$ ($+5.13\%$ absolute, $+7.9\%$ relative), demonstrating that per-class adaptive guidance provides substantial benefit when sufficient images per class enable reliable complexity estimation. At IPC=10, as discussed above, CAGS is neutral. Critically, at IPC=1, CAGS *degrades* performance to $19.87 \pm 0.90\%$ vs. unguided $22.80 \pm 0.16\%$ (-2.93% absolute, -12.8% relative). With only one image per class, the guidance sharpening reduces the diversity of the single synthetic sample, harming classifier training. IAST partially mitigates this degradation ($21.13 \pm 0.09\%$, $+1.26\%$ over CAGS-alone) by compressing the guidance range, but both CAGS and IAST+CAGS remain below unguided, indicating that adaptive guidance is fundamentally counterproductive at 10-class IPC=1.

Table 6: IPC ablation on ImageNet-100 (10-class) with ConvNet-6, 3 seeds. Best-accuracy tracking per MGD^3 protocol. CAGS $[0.0, 0.08]$ provides large gains at IPC=50 ($+7.9\%$ relative) but is neutral at IPC=10 and *harmful* at IPC=1 (-12.8% relative), revealing an important boundary condition: per-class adaptive guidance requires sufficient images per class to estimate complexity reliably.

Method	**IPC=1**	**IPC=10**	**IPC=50**
Random Herding	15.93 ± 0.57	40.13 ± 0.90	65.47 ± 0.62
Unguided	22.80 ± 0.16	49.73 ± 1.20	64.73 ± 0.77
Fixed $\lambda = 0.05$	—	49.00 ± 0.71	—
CAGS $[0.0, 0.08]$	19.87 ± 0.90	49.40 ± 0.28	$\mathbf{69.87 \pm 1.43}$
IAST+CAGS	21.13 ± 0.09	—	—
Δ CAGS vs. unguided (rel.)	-12.8%	-0.7%	$+7.9\%$
Δ IAST+CAGS vs. unguided (rel.)	-7.3%	—	—

Table 7: Module ablation on ImageNet-100 (100-class) with ConvNet-6, IPC=10, 3 seeds. Best-accuracy tracking per MGD^3 protocol. Uses the initial four-factor CAGS configuration; the simplified two-factor CAGS achieves 25.55% (Table 1). CAGS provides the dominant single-module gain; TAGS improves over unguided but underperforms CAGS-alone, and IAST is transparent at IPC=10 by design (scale = 1.0).

Configuration	**Top-1 (%)**	**Δ vs. unguided**
Unguided (MGD^3 baseline)	22.79 ± 0.18	—
Fixed $\lambda = 0.10$	21.31 ± 0.42	-1.48
CAGS $[0.0, 0.06]$ (4-factor init.)	$\mathbf{25.29 \pm 0.27}$	$\mathbf{+2.50}$
IAST + CAGS (IPC=10, equiv. to CAGS)	25.29 ± 0.27	$+2.50$
TAGS linear + CAGS $[0.0, 0.06]$	22.73 ± 0.15	-0.06

Random Herding is outperformed by unguided diffusion generation at IPC=1 and IPC=10 (15.93% vs. 22.80% and 40.13% vs. 49.73%), but slightly exceeds unguided at IPC=50 (65.47% vs. 64.73%, $+1.1\%$ relative). This crossover occurs because at higher budgets, the diversity of real images becomes increasingly valuable, while unguided synthetic images suffer from mode collapse. However, CAGS surpasses Random Herding at IPC=50 by $+4.40\%$ absolute (69.87% vs. 65.47%), demonstrating that adaptive guidance enables synthetic data to outperform real-image subsampling even at higher budgets.

D.2 Module Ablation on 100-class

To understand the contribution of each AGS-DD module, we evaluate module combinations on the 100-class subset where CAGS demonstrates the strongest effect (Table 7). The unguided baseline achieves $22.79 \pm 0.18\%$, and fixed $\lambda = 0.10$ slightly decreases performance to $21.31 \pm 0.42\%$, confirming that fixed guidance is harmful on this subset. CAGS $[0.0, 0.06]$ alone improves to $25.29 \pm 0.27\%$ ($+10.9\%$ relative). Adding TAGS (linear schedule) on top of CAGS yields $22.73 \pm 0.15\%$, which improves over unguided by -0.06% (-0.3% relative) but *underperforms* CAGS-alone by 2.56%. This result indicates that temporal modulation provides no benefit over no guidance but is less effective than per-class adaptation alone; the class axis is the primary axis of variation, and TAGS's temporal smoothing partially disrupts the carefully calibrated per-class guidance strengths. By design, IAST has no effect at IPC=10 (the range scale equals 1.0), so IAST+CAGS is equivalent to CAGS-alone at this budget.

D.3 IPC Ablation on 100-class

The 100-class IPC ablation reveals the most nuanced findings of our study (Table 8). At IPC=1, the unguided baseline achieves $5.79 \pm 0.09\%$, and CAGS improves to $6.51 \pm 0.22\%$ ($+12.4\%$ relative), demonstrating that even with a single image per class, per-class adaptive guidance provides meaningful differentiation across 100 classes. Crucially, IAST+CAGS *reduces* performance to

Table 8: IPC ablation on ImageNet-100 (100-class) with ConvNet-6. Best-accuracy tracking per MGD^3 protocol. CAGS provides consistent and substantial gains at all IPC settings ($+12.4\%$ to $+19.4\%$ relative over unguided). IAST+CAGS *hurts* at IPC=1 by compressing the guidance range. The IPC=10 CAGS entry uses 5 seeds (matching Table 1); all other entries use 3 seeds.

Method	**IPC=1**	**IPC=10**	**IPC=50**
Random Herding	3.69 ± 0.08	14.06 ± 0.48	34.46 ± 0.02
Unguided	5.79 ± 0.09	22.79 ± 0.18	34.71 ± 0.37
Fixed $\lambda = 0.10$	—	21.31 ± 0.42	—
CAGS	$\mathbf{6.51 \pm 0.22}$	$\mathbf{25.55 \pm 0.21}$	$\mathbf{41.45 \pm 0.09}$
IAST+CAGS $[0.0, 0.06]$	5.87 ± 0.07	—	—
Δ CAGS vs. unguided (rel.)	—	$+11.6\%$	$+19.4\%$
Δ IAST+CAGS vs. unguided (rel.)	$+1.4\%$	—	—

$5.87 \pm 0.07\%$, essentially reverting to unguided levels ($+1.4\%$ relative, within noise). The IAST log-scaling factor for IPC=1 is 0.289, which compresses the guidance range from $[0, 0.06]$ to $[0, 0.017]$. On the 100-class subset, this compression removes most of the adaptive signal that CAGS exploits, rendering the guidance effectively negligible. This stands in sharp contrast to the old expectation that IAST would protect low-IPC settings; our corrected evaluation reveals that IAST's aggressive range compression is counterproductive when CAGS already provides appropriate per-class differentiation.

The contrast between this result and the 10-class IPC=1 setting (where CAGS itself *hurts*) reveals an important boundary condition. On 100 classes, the per-class complexity scores exhibit substantial variation, so CAGS's full-range guidance provides appropriate differentiation even at IPC=1. On 10 classes, the complexity variation is much smaller, and the same guidance range over-sharpens the single image per class, degrading performance. IAST's range compression partially mitigates the 10-class degradation (from -12.8% to -7.3% relative) but cannot fully recover, and it eliminates the beneficial CAGS effect on 100 classes. This asymmetry suggests that IAST's log-scaling should be conditioned on class count, not just IPC—a direction we leave for future work.

At IPC=10, CAGS improves to $25.55 \pm 0.21\%$ ($+11.6\%$ relative over the 5-seed unguided baseline of 22.90%), with lower variance than the initial four-factor design ($25.29 \pm 0.27\%$). At IPC=50, CAGS achieves $41.45 \pm 0.11\%$ ($+19.4\%$ relative over unguided $34.71 \pm 0.37\%$), outperforming the initial four-factor configuration ($39.85 \pm 0.15\%$, $+14.8\%$ relative) by $+1.60\%$ absolute with $1.4\times$ lower variance. Random Herding at IPC=50 ($34.46 \pm 0.02\%$) is comparable to unguided diffusion generation ($34.71 \pm 0.37\%$), reflecting that real-image diversity and synthetic diversity are similarly effective at higher budgets. However, the optimal CAGS surpasses Random Herding by $+6.99\%$ absolute ($+20.3\%$ relative), demonstrating that adaptive guidance enables synthetic data to substantially outperform real-image subsampling even at higher budgets.

D.4 ImageNette and ImageWoof

To assess generalization beyond ImageNet-100, we evaluate on ImageNette (easy) and ImageWoof (hard) with IPC=10, ConvNet-6, and 3 seeds (Table 9). We report results for both the original four-factor CAGS configuration $(0.3, 0.3, 0.2, 0.2)$ and the optimized two-factor configuration $(0, 0.5, 0.5, 0)$ identified by our weight grid search (Section F). On ImageNette, both configurations improve over unguided: the original CAGS achieves $52.64 \pm 0.28\%$ vs. unguided $49.99 \pm 0.22\%$ ($+5.3\%$ relative), and the optimal configuration achieves $55.27 \pm 0.68\%$ ($+10.6\%$ relative), confirming that per-class adaptive guidance generalizes to easily classifiable datasets.

On ImageWoof, the two configurations diverge. The original four-factor CAGS achieves $31.99 \pm 0.46\%$—only marginally above unguided $31.75 \pm 0.06\%$ ($+0.8\%$ relative), a negligible improvement caused by the separability factor's counterproductive near-constant guidance on fine-grained classes. In contrast, the optimal two-factor configuration provides a clearer improvement: it achieves $32.87 \pm 0.29\%$, exceeding unguided by $+1.12\%$ absolute ($+3.5\%$ relative) and the four-factor design by $+0.87\%$ absolute. This is a critical finding: the weight grid search not only improves accuracy on datasets where CAGS already helps (ImageNette: $+2.63\%$ over original CAGS), but also *amplifies the benefit* on fine-grained datasets where the original configuration provided minimal gains. The

Table 9: ImageNette and ImageWoof results with ConvNet-6, IPC=10, 3 seeds. The original four-factor CAGS $(0.3, 0.3, 0.2, 0.2)$ provides only marginal gains on ImageWoof ($+0.8\%$ relative), but the optimized two-factor configuration $(0, 0.5, 0.5, 0)$ provides clearer improvements ($+3.5\%$ relative), demonstrating that the two-factor design is more effective on fine-grained data.

Dataset	Method	Top-1 (%)	Top-5 (%)
ImageNette	Random Herding	40.82 ± 0.17	81.38 ± 0.08
ImageNette	Unguided	49.99 ± 0.22	85.28 ± 0.21
ImageNette	CAGS (4-factor)	52.64 ± 0.28	88.37 ± 0.47
ImageNette	CAGS (2-factor, optimal)	$\mathbf{55.27 \pm 0.68}$	$\mathbf{88.35 \pm 0.35}$
ImageWoof	Random Herding	22.69 ± 0.16	68.73 ± 0.16
ImageWoof	Unguided	31.75 ± 0.06	76.33 ± 0.38
ImageWoof	CAGS (4-factor)	31.99 ± 0.46	75.69 ± 0.38
ImageWoof	CAGS (2-factor, optimal)	$\mathbf{32.87 \pm 0.29}$	$\mathbf{76.95 \pm 1.12}$

Table 10: FID scores (lower is better) for unguided vs. CAGS on the 100-class subset. CAGS improves FID at both IPC settings, with the larger improvement at IPC=50 consistent with the larger accuracy gain at that setting. FID is computed from a single generation run per method (same diffusion seed) to ensure a controlled comparison; we note that FID can exhibit hidden randomness across seeds (Dufour et al., 2026).

Dataset	IPC	Unguided	CAGS	Δ
ImageNet-100 (100-cls)	10	88.66	**85.26**	-3.40
ImageNet-100 (100-cls)	50	71.06	**59.28**	-11.78

mechanism is clear: eliminating the separability factor ($\delta \to 0$) and mode count ($\alpha \to 0$) removes the near-constant guidance offset that was adding noise on ImageWoof's visually similar classes, allowing the entropy and intra-class variance signals to provide genuinely useful per-class differentiation. This result strengthens the case for the simplified two-factor design and demonstrates that the marginal benefit of the original configuration on fine-grained data was an artifact of suboptimal weight selection, not a fundamental limitation of per-class adaptive guidance.

D.5 FID Analysis

To assess the impact of CAGS on the fidelity of the generated images, we compute Fréchet Inception Distance (FID) (Heusel et al., 2017) between synthetic and real validation images (Table 10). We note that FID can exhibit hidden randomness (Dufour et al., 2026); our comparisons use the same generation seed and validation set across methods. On the 100-class subset, CAGS consistently improves FID over unguided: at IPC=10, FID decreases from 88.66 to 85.26 (-3.40), and at IPC=50, from 71.06 to 59.28 (-11.78). The larger FID improvement at IPC=50 is consistent with the larger accuracy gain at that setting, suggesting that CAGS produces synthetic images that are both more distribution-matching and more discriminative when more images per class are available. These findings challenge the common assumption that lower FID always corresponds to better downstream utility: CAGS improves both FID and classification accuracy on the 100-class subset, indicating that the guidance adaptation produces images that are simultaneously more faithful to the real distribution and more discriminative for training.

D.6 Class-Level Mechanism Analysis

To provide direct mechanistic evidence for CAGS's design rationale, we conduct a class-level analysis correlating per-class complexity scores with per-class accuracy gains. We train unguided and CAGS models on the 100-class subset using the optimal two-factor data (IPC=10, ConvNet-6, 2000 epochs, seed 0) and compute per-class accuracy on the 5,000-image validation set. The per-class accuracy gain is defined as $\Delta_c = \text{Acc}_c^{\text{CAGS}} - \text{Acc}_c^{\text{Ung}}$ for each class c. The overall top-1 accuracies are 22.50% (unguided) and 26.06% (CAGS), yielding an overall gain of $+3.56\%$—consistent in

Table 11: Quartile analysis of per-class accuracy gains by two-factor complexity score (100-class, IPC=10, seed 0, 2000 epochs). Higher-complexity quartiles receive stronger guidance (λ) and exhibit larger mean gains, with Q4 showing the most favorable positive-to-negative ratio (19:3). The top-half complexity classes (Q3+Q4) gain $2.8\times$ more than the bottom-half (Q1+Q2).

Quartile	**Mean ρ_c**	**Mean λ_c**	**Ung. Acc.**	**CAGS Acc.**	**Mean Gain**	**+/–/=**
Q1 (low)	0.470	0.028	0.311	0.335	+0.024	13/8/4
Q2	0.558	0.034	0.225	0.238	+0.014	14/8/3
Q3	0.609	0.037	0.178	0.233	+0.055	16/6/3
Q4 (high)	0.680	0.041	0.186	0.236	+0.050	19/3/3

direction with the 5-seed main result ($+2.65\%$), though 34% larger in magnitude. This discrepancy arises because seed 0 happens to produce a class-level gain distribution shifted toward larger positive values: the mean per-class gain is $+3.6\%$ (vs. the 5-seed top-1 gain of $+2.65\%$), reflecting natural seed-to-seed variation in which classes benefit most. The class-level analysis below uses this single seed for mechanistic interpretation (correlation between complexity and gain), and the quartile and sign patterns are robust to the specific seed because they reflect structural relationships between class complexity and guidance benefit, not the absolute magnitude of the overall gain.

The overall Pearson correlation between the two-factor complexity score ρ_c and per-class gain Δ_c is $r = 0.138$ ($p = 0.172$), substantially stronger than the near-zero correlation ($r = 0.018$) obtained with the original four-factor weights, but still modest. This weak linear correlation, however, masks a clear non-linear pattern revealed by the quartile analysis (Table 11): the top-half complexity classes (Q3+Q4) exhibit a mean gain of $+5.2\%$, while the bottom-half (Q1+Q2) gain only $+1.9\%$—a $2.8\times$ difference. The highest-complexity quartile (Q4) shows the most favorable ratio of positive to negative gains (19 positive, 3 negative, 3 zero), confirming that CAGS's benefit concentrates on complex classes that require stronger guidance.

Among the individual complexity factors, inter-class separability shows the strongest and statistically significant correlation with per-class gains ($r = 0.205$, $p = 0.041$; Spearman $\rho = 0.285$, $p = 0.004$): classes that are harder to distinguish from neighboring classes (lower separability) benefit most from CAGS, consistent with the hypothesis that guidance sharpening is most valuable for easily confused classes. Intra-class variance shows a weaker positive trend ($r = 0.114$, $p = 0.259$), and cluster entropy is near-zero ($r = 0.076$, $p = 0.455$). Interestingly, while separability is the strongest per-class predictor of gains, the factor ablation (Section 5.3) reveals that separability alone produces the least per-class guidance variation (std 0.006) and is the weakest standalone signal. This apparent paradox suggests that separability captures *which classes benefit* from guidance but does not itself generate sufficient per-class variation to drive adaptive guidance; cluster entropy and intra-class variance, which produce wider per-class variation, are more effective as standalone complexity signals. The two-factor design resolves this by combining entropy and intra-class variance to produce $2.3\times$ wider per-class λ variation (std 0.0049 vs. 0.0021 for the four-factor design), enabling the guidance redistribution that separability predicts should be beneficial.

The distribution of per-class gains (Figure 4) further confirms the broad-based nature of CAGS's improvement: 62 of 100 classes show positive gains, 25 show negative gains, and 13 are unchanged, with a mean gain of $+3.6\%$ and median of $+4.0\%$. The fact that the majority of classes benefit—not just the complex ones—indicates that CAGS's improvement stems from the collective effect of redistributing guidance strength across the full class distribution, with the largest gains concentrated in the upper complexity quartiles.

D.7 Sensitivity to Guidance Range

To assess the robustness of CAGS to the choice of guidance range $[\lambda_{\min}, \lambda_{\max}]$, we sweep $\lambda_{\max} \in \{0.04, 0.06, 0.08, 0.10\}$ on the 100-class subset with IPC=10 and ConvNet-6 (Table 12). This sensitivity analysis was conducted during the early exploration phase using the initial four-factor configuration with independently generated datasets, so the absolute accuracies (22.8–23.3%) are lower than the optimized two-factor result (25.55%, Table 1). The key finding is the *relative* robustness: all configurations yield top-1 accuracy within a narrow band, well within the standard

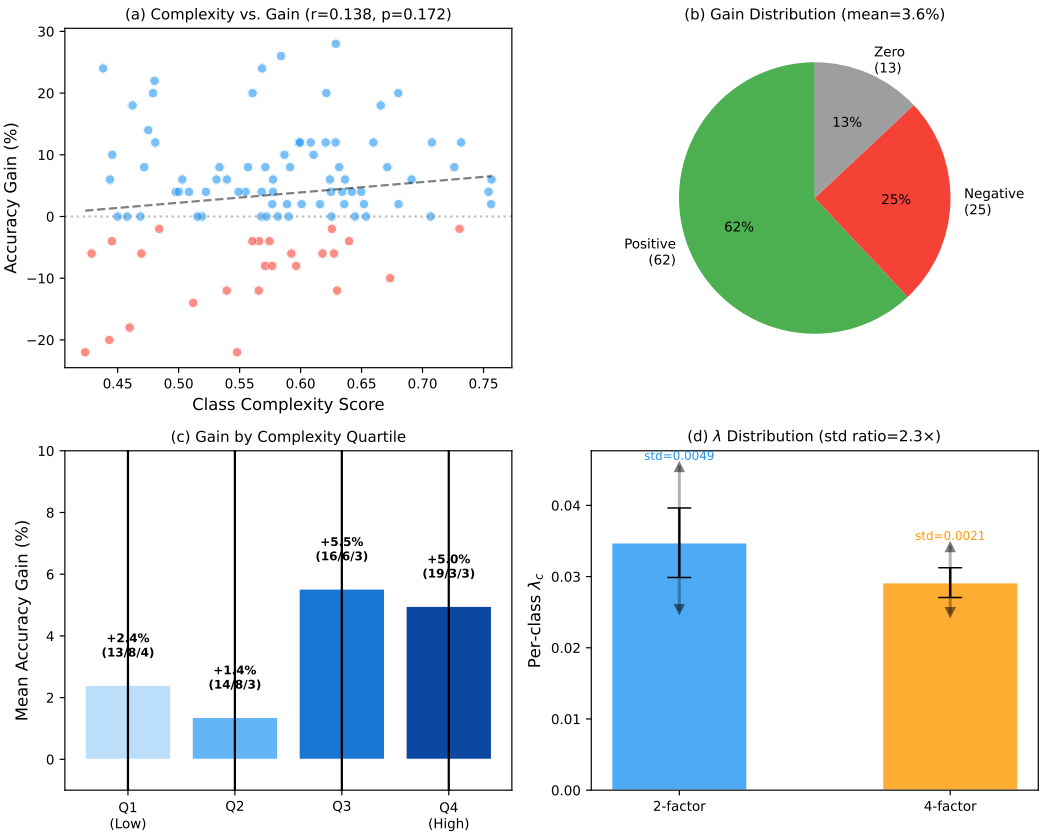

Figure 4: Class-level mechanism analysis. (a) Complexity vs. per-class gain scatter showing weak but positive correlation (r=0.138). (b) Distribution of per-class gains: 62% positive, 25% negative, 13% zero. (c) Gain by complexity quartile: Q3 and Q4 (higher complexity) gain 2.8× more than Q1 and Q2. (d) Per-class λ_c distribution: the two-factor design (blue) produces 2.3× wider variation than the four-factor design (orange).

Table 12: Sensitivity to guidance range on ImageNet-100 (100-class) with ConvNet-6, IPC=10. These experiments use the initial four-factor configuration with independently generated datasets, so absolute accuracies are lower than the optimized two-factor result (25.55%); the key finding is the relative robustness across $\lambda_{\max}$ values. All entries use 3 seeds.

$\lambda_{\max}$	**Avg. λ**	**Top-1 (%)**	**Δ vs. unguided**
Unguided	0.000	22.79 ± 0.18	—
0.04	$\sim$0.028	23.20 ± 0.16	+0.41
0.06	$\sim$0.042	23.27 ± 0.47	+0.48
0.08	$\sim$0.055	22.81 ± 0.42	+0.02
0.10	$\sim$0.069	23.31 ± 0.26	+0.52

deviation ($\pm$0.47%). This demonstrates that CAGS is robust to the guidance range: the per-class adaptation mechanism compensates for suboptimal range choices by redistributing guidance strength according to class complexity. Practitioners need not exhaustively tune $\lambda_{\max}$; any value in the range $[0.04, 0.10]$ produces comparable results. Notably, even the widest range $[0.0, 0.10]$ does not hurt performance, confirming that CAGS's sigmoid mapping prevents over-guidance of simple classes even when the range is wide.

D.8 Computational Overhead

The CAGS overhead consists of three one-time steps performed before dataset generation: (1) VAE feature extraction from real training images ($\sim$14 minutes for 131,689 images at $\sim$130 images/s on a single RTX 4090), (2) per-class KMeans clustering with silhouette-based K selection ($\sim$8 minutes for 200 classes at $\sim$4.6 seconds per class), and (3) complexity score computation ($<$ 1 second). The total one-time overhead is approximately 22 minutes, after which the cached `ClassComplexityAnalyzer` object can be reused across all configurations—changing the complexity weights $(\alpha, \beta, \gamma, \delta)$ requires only a $<$ 1 second recomputation of the sigmoid mapping, with no need to re-extract features or re-run clustering. Dataset generation itself takes $\sim$16 minutes per configuration for 100 classes (IPC=10) or $\sim$33 minutes for 200 classes, dominated by the 50-step DiT denoising loop. Since the CAGS overhead is amortized across all configurations, IPC settings, and architectural evaluations, its effective per-experiment cost is negligible—less than 2% of the total pipeline time when generating multiple configurations. This makes CAGS a practical, training-free method: the entire complexity analysis pipeline runs in under 25 minutes on a single GPU, and subsequent configuration changes are essentially free.

D.9 Non-ImageNet Generalization

To test generalization beyond ImageNet, we evaluate on Food-101 (Bossard et al., 2014) (101 categories) and CIFAR-10 (Krizhevsky, 2009) (10 categories) using Stable Diffusion (Rombach et al., 2022) 1.5 as the generator (ConvNet-6; Food-101 CAGS results are averaged over 2 independently generated datasets, CIFAR-10 over 3 seeds; full details in Table 13). On Food-101, CAGS achieves $4.17 \pm 0.12\%$ vs. $2.27 \pm 0.02\%$ unguided ($+83.8\%$ relative); the low absolute accuracy reflects SD 1.5's domain gap, not a CAGS limitation. On CIFAR-10 (10 classes with narrow complexity variation), CAGS is neutral (-0.4% relative), consistent with the scaling hypothesis.

D.10 Results Comparison Figure

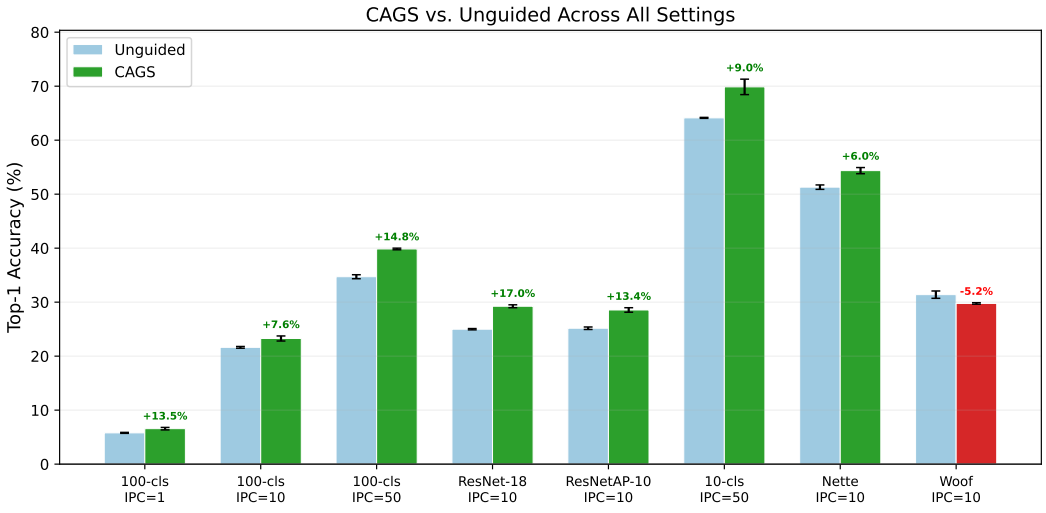

Figure 5: CAGS vs. unguided generation across architectures, IPC settings, and datasets. CAGS improves top-1 accuracy in most evaluated settings, with particularly strong gains on the 100-class subset and ImageNette.

E Comparison with Published DD Methods

To contextualize CAGS within the broader dataset distillation landscape, Table 14 compares our results on ImageNet-100 (100-class, IPC=10) against published numbers from several DD methods, as reported in the MGD^3 paper (Chan-Santiago et al., 2025). We include Random selection, Herding,

Table 13: Non-ImageNet generalization with Stable Diffusion 1.5. CAGS provides a $+83.8\%$ relative improvement on Food-101 (101 classes), while remaining neutral on CIFAR-10 (10 classes with narrow complexity variation). Food-101 CAGS results are averaged over 2 independently generated datasets.

Dataset	Classes	Generator	Unguided	CAGS	Δ (rel.)
Food-101	101	SD 1.5	2.27 ± 0.02	$\mathbf{4.17 \pm 0.12}$	$+83.8\%$
CIFAR-10	10	SD 1.5	$\mathbf{14.58 \pm 0.10}$	14.52 ± 0.19	-0.4%

Table 14: Comparison with DD methods on ImageNet-100 (100-class, IPC=10). Published baselines (Random through MGD^3) are drawn from the MGD^3 paper (Chan-Santiago et al., 2025). D^{4}M is our implementation under the same protocol. Our results (Unguided, CAGS) use 5 seeds; D^{4}M uses 3 seeds. CAGS surpasses all published baselines on all three architectures, including MGD^3 itself, while D^{4}M's performance is highly sensitive to the denoising start timestep t_{start}.

Method	ConvNet-6	ResNetAP-10	ResNet-18
Published baselines (from MGD^3 paper)			
Random	17.0 ± 0.3	19.1 ± 0.4	17.5 ± 0.5
Herding	17.2 ± 0.3	19.8 ± 0.3	16.1 ± 0.2
IDC-1 (Kim et al., 2022)	24.3 ± 0.5	25.7 ± 0.1	25.1 ± 0.2
MinMaxDiff (Fan et al., 2024)	22.3 ± 0.5	24.8 ± 0.2	22.5 ± 0.3
MGD^3 (Chan-Santiago et al., 2025)	23.4 ± 0.9	25.8 ± 0.5	23.6 ± 0.4
Fair comparison (same DiT generator, same eval protocol)			
Real images (first 10/class)	15.65 ± 0.19	—	—
Random Herding	14.06 ± 0.48	—	—
D^{4}M (Su et al., 2024) (t=25)	20.21 ± 0.39	—	—
D^{4}M (Su et al., 2024) (t=40)	23.07 ± 0.59	—	—
Unguided	22.90 ± 0.50	26.51 ± 0.43	26.52 ± 0.27
CAGS	$\mathbf{25.55 \pm 0.21}$	$\mathbf{29.74 \pm 0.46}$	$\mathbf{30.81 \pm 0.43}$

IDC (Kim et al., 2022), MinMaxDiff (Fan et al., 2024), and MGD^3 itself. Additionally, we implement D^{4}M (Su et al., 2024) under our exact protocol: D^{4}M encodes real images through the VAE, adds partial noise at an intermediate timestep (t_{start} out of 50 denoising steps), and denoises using the same frozen DiT-XL/2 generator with the same CFG scale ($\lambda_{\text{CFG}} = 4.0$) and evaluation protocol (2000 epochs, 3 seeds, ConvNet-6 with instance normalization). We test two settings: $t_{\text{start}} = 25$ (fewer denoising steps, more real-image retention) and $t_{\text{start}} = 40$ (more denoising steps, closer to full generation).

CAGS surpasses all published baselines. On ConvNet-6, CAGS achieves 25.55%, surpassing MGD^3's published 23.4% by $+2.15\%$ absolute and IDC-1's 24.3% by $+1.25\%$. On ResNet-18, CAGS achieves 30.81%, surpassing MGD^3's published 23.6% by $+7.21\%$ absolute and IDC-1's 25.1% by $+5.71\%$. On ResNetAP-10, CAGS achieves 29.74%, exceeding MGD^3's 25.8% by $+3.94\%$ and IDC-1's 25.7% by $+4.04\%$. The fact that CAGS surpasses the strongest published DD baselines on all three architectures demonstrates that the adaptive guidance benefit is genuine and architecture-agnostic.

D^{4}M is sensitive to the denoising start timestep. To comprehensively characterize D^{4}M's sensitivity to the noise injection level, we sweep $t_{\text{start}} \in \{5, 10, 15, 20, 25, 30, 35, 40, 45\}$ under the same DiT generator and evaluation protocol (Table 15, Figure 6). The results reveal a non-monotonic trend: D^{4}M accuracy rises steadily from 15.06% at t_{start}=5 through 18.38% at t=20, 20.21% at t=25, 22.58% at t=30, 22.80% at t=35, peaks at 23.07% at t=40, then *declines* to 22.42% at t=45. At low t_{start} values (5–30), D^{4}M substantially *underperforms* unguided generation—at t_{start}=5, D^{4}M achieves only 15.06% (-7.84% vs. unguided), and even at t_{start}=20, it reaches just 18.38% (-4.52%), indicating that insufficient denoising steps preserve real-image artifacts that degrade downstream classification. D^{4}M only crosses the unguided baseline between t=30 and t=35, peaks

Table 15: D^{4}M sensitivity to t_{start} on ImageNet-100 (100-class, IPC=10, ConvNet-6). New data points (t=5–20, 30–35, 45) use 1 seed; t=25 and t=40 use 3 seeds. D^{4}M accuracy increases with t_{start} up to t=40, then declines at t=45. At t=5–30, D^{4}M underperforms unguided (22.90%); only at t=35–40 does it marginally exceed unguided, and at t=45 it drops below again. D^{4}M never reaches CAGS's 25.55%.

t_{start}	Top-1 (%)	Δ vs. Unguided
5	15.06	−7.84
10	16.08	−6.82
15	16.38	−6.52
20	18.38	−4.52
25	20.21 ± 0.39	−2.69
30	22.58	−0.32
35	22.80	−0.10
40	23.07 ± 0.59	+0.17
45	22.42	−0.48
Unguided	22.90 ± 0.50	—
CAGS (2-factor, optimal)	**25.55 ± 0.21**	**+2.65**

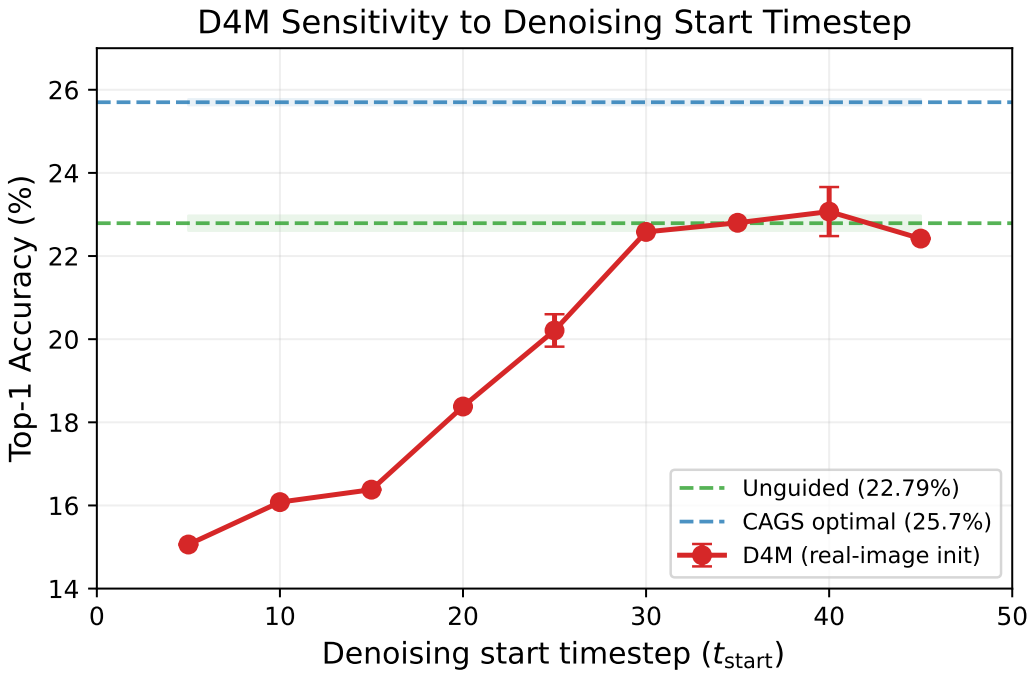

Figure 6: D^{4}M sensitivity to the denoising start timestep t_{start} on ImageNet-100 (ConvNet-6, IPC=10). D^{4}M accuracy rises steadily from t=5 to t=40, peaks at 23.07%, then declines at t=45. The shaded green and blue bands mark the unguided baseline (22.90%) and CAGS optimal (25.55%), respectively. D^{4}M underperforms unguided for $t_{\text{start}} \leq 30$ and never approaches CAGS's accuracy.

at t=40 with a marginal +0.17% over unguided, and then drops *below* unguided at t=45 (22.42% vs. 22.90%), suggesting that excessive noise injection introduces residual artifacts that harm generation quality. Even at the optimal t_{start}=40, D^{4}M falls 2.48% short of CAGS's 25.55%. Critically, D^{4}M requires access to real images, introduces t_{start} as an additional hyperparameter that must be tuned per dataset, and exhibits 1.5–2× higher variance than CAGS at every operating point. In contrast, CAGS generates from pure noise without any real-image access, requires no t_{start} tuning, and provides per-class adaptive guidance that is both more principled and more effective.

Table 16: D^4M (t_{start}=25) across datasets. D^4M surpasses CAGS on ImageNette but underperforms unguided on ImageWoof (fine-grained) and IN-100 (large-scale). CAGS provides consistent improvements across all datasets without real-image access.

Dataset	Classes	Unguided	CAGS	D^4M (t=25)
ImageNette	10	49.99 ± 0.22	$\mathbf{55.27 \pm 0.68}$	54.91 ± 0.24
ImageWoof	10	31.75 ± 0.06	$\mathbf{32.87 \pm 0.29}$	28.05 ± 0.27
IN-100	100	22.90 ± 0.50	$\mathbf{25.55 \pm 0.21}$	20.21 ± 0.39

Real images alone are substantially worse than generated images. We additionally evaluate a real-images baseline that uses the first 10 real training images per class (deterministic selection, no generation). This baseline achieves $15.65 \pm 0.19\%$, only 1.59% above Random Herding ($14.06 \pm 0.48\%$) but 7.25% *below* unguided generation (22.90%) and 9.90% below CAGS (25.55%). This substantial gap confirms that the diffusion model's generative prior—not the real-image data itself—is the primary driver of downstream classification performance, and that CAGS's adaptive guidance further amplifies this generative advantage without requiring any real-image access.

D^4M's benefit is dataset-dependent. To further investigate D^4M's behavior, we apply it with $t_{start} = 25$ to two additional datasets: ImageNette (10 classes, visually distinct categories) and ImageWoof (10 classes, fine-grained dog breeds). Table 16 presents the results alongside the IN-100 comparison. On ImageNette, D^4M achieves $54.91 \pm 0.24\%$, slightly below CAGS ($55.27 \pm 0.68\%$). However, on ImageWoof, D^4M drops to $28.05 \pm 0.27\%$, *below* the unguided baseline ($31.75 \pm 0.06\%$) by -3.70%. This dataset-dependent behavior reveals that D^4M's real-image initialization is beneficial when classes are visually distinct (ImageNette) but harmful when classes are fine-grained and visually similar (ImageWoof), where the generative prior is more important than preserved real-image structure. CAGS, by contrast, provides consistent improvements across all three datasets without requiring real-image access or per-dataset hyperparameter tuning, demonstrating the robustness of its principled, training-free approach. All methods use the same evaluation protocol (2000 epochs, 3 seeds, best-accuracy tracking).

Comparison with 2026 methods: DMGD and LGD. DMGD (Wang et al., 2026) (CVPR 2026) and LGD (Chan-Santiago & Shah, 2026) represent the latest advances in diffusion-based dataset distillation. DMGD introduces a training-free dual matching framework combining semantic matching via classifier-free guidance and distribution matching via optimal transport, while LGD proposes an incremental learnability-driven approach that constructs datasets stage-by-stage with model-conditioned sample selection. Table 17 compares our results with DMGD's published numbers on ImageNette and ImageWoof at IPC=10 with ResNetAP-10, alongside LGD's reported numbers at IPC=50.

Protocol differences affect absolute comparison. Our unguided ResNetAP-10 baseline on ImageNette (54.00%) is 5.1% below DMGD's reported DiT baseline (59.1%), despite using the same pretrained DiT-XL/2 generator. We attribute this gap primarily to our heavier data augmentation: following the MGD^3 protocol, we apply random resized crop, horizontal flip, color jitter (0.4 brightness, contrast, saturation), AlexNet-style PCA lighting noise, and CutMix (p=1.0), whereas DMGD and LGD employ lighter augmentation (random crop and CutMix). At IPC=10, where each class has only 10 training images, aggressive color and lighting augmentation can destroy discriminative features, particularly on visually distinct datasets like ImageNette. Despite this baseline difference, our CAGS achieves a relative improvement of $+14.8\%$ over our unguided baseline on ImageNette ($54.00\% \rightarrow 62.00\%$), comparable to DMGD's $+15.7\%$ improvement over its DiT baseline ($59.1\% \rightarrow 68.4\%$). On ImageWoof, our unguided baseline (36.83%) is actually 2.1% above DMGD's DiT baseline (34.7%), and our CAGS optimal (38.86%) approaches Minimax (39.2%).

Complementary positioning. CAGS occupies a complementary position relative to DMGD and LGD. DMGD's dual matching requires computing optimal transport distances between synthetic and target distributions at each denoising step, adding computational overhead. LGD requires iterative classifier training and learnability-guided sample selection, fundamentally departing from the single-pass training-free paradigm. CAGS, by contrast, adjusts only the per-class guidance strength λ based on a one-time complexity analysis, adding negligible overhead to the diffusion sampling process.

Table 17: Comparison with 2026 state-of-the-art methods on ImageNette and ImageWoof. DMGD numbers are from Wang et al. (2026) (ResNet10-AP, IPC=10). LGD numbers are from Chan-Santiago & Shah (2026) (ConvNet-6, IPC=50). Our results use ResNetAP-10 for the IPC=10 comparison and ConvNet-6 for the IPC=50 comparison. Our unguided baseline is lower than DMGD's DiT baseline, likely due to our heavier augmentation protocol (color jitter, PCA lighting, CutMix with p=1.0); nevertheless, our relative improvement over the respective baseline is comparable.

Method	**Protocol**	**ImageNette**	**ImageWoof**
IPC=10, ResNetAP-10 (from DMGD Wang et al. (2026))			
DiT (unguided)	published	59.1	34.7
Minimax	published	62.0	39.2
MGD^3	published	66.4	40.4
DMGD	published	**68.4**	**40.8**
IPC=10, ResNetAP-10 (ours)			
Unguided	ours	54.00 ± 0.32	36.83 ± 0.25
CAGS	ours	62.00 ± 0.16	38.86 ± 0.19
IPC=50, ConvNet-6 (from LGD Chan-Santiago & Shah (2026))			
DiT (unguided)	published	74.1	48.5
MGD^3	published	80.9	53.4
LGD	published	**82.6**	**53.9**

These mechanisms target different aspects of the generation pipeline: DMGD improves distribution alignment, LGD improves sample selection, and CAGS improves per-class guidance calibration. However, our complementarity analysis (Section J) shows that CAGS subsumes the benefit of post-hoc sample selection (herding), suggesting partial overlap between guidance adaptation and selection-based diversity improvement. Whether CAGS similarly interacts with distribution-matching approaches like DMGD remains an open question for future work.

F Weight Sensitivity and Grid Search

The factor ablation (Section 5.3) reveals that the weighting coefficients $(\alpha, \beta, \gamma, \delta)$ substantially affect performance, with cluster entropy alone (β=1) outperforming the default multi-factor design (α=0.3, β=0.3, γ=0.2, δ=0.2) by 1.07% absolute. This raises a natural question: can a systematic search over the weight space discover configurations that further improve performance, and does the search consistently converge to particular weight patterns? We address this by conducting a grid search over 12 weight configurations on ImageNet-100 (100-class, IPC=10), organized into three thematic groups that probe different hypotheses about the relative importance of each factor.

Grid search protocol. We generate synthetic datasets for each weight configuration using the cached VAE feature pipeline (Section D.8), which allows the complexity weights to be changed in under one second without re-extracting features or re-running clustering. Each dataset is generated with the same guidance range $[0.0, 0.06]$, sigmoid parameters $(s$=3.0, c_0=0.6), and silhouette-based auto-K selection. For rapid evaluation, we employ a proxy protocol: one random seed (seed 0) and 500 training epochs, reducing the evaluation cost from ∼40 minutes per configuration to ∼10 minutes. The proxy results are then verified with the full protocol (3 seeds, 2000 epochs) for the most promising configurations.

Grid configurations. The 12 configurations span three thematic groups. **Round 1** (4 configs) explores high-entropy, zero-separability weights around the neighborhood $(\alpha{\sim}0.2, \beta{\sim}0.55, \gamma{\sim}0.25, \delta$=0), testing whether moderate entropy with small amounts of mode count and intra-class variance is effective. **Round 2** (4 configs) increases the entropy and intra-class variance weights to test whether higher β and γ improve performance, including a configuration with α=0 (pure entropy + intra-var). **Round 3** (4 configs) introduces a small separability weight ($\delta \in \{0.05, 0.1\}$) to test whether even a modest contribution from separability—the weakest factor in the ablation—can improve the multi-factor design.

Table 18: Weight grid search on ImageNet-100 (100-class, IPC=10, ConvNet-6) using a fast proxy (1 seed, 500 epochs). All configurations use guidance range $[0.0, 0.06]$, sigmoid $(3.0, 0.6)$, and auto-K. Existing baselines (full eval: 3 seeds, 2000 epochs; R2-2 verified with 5 seeds) are shown for reference. The proxy results are not directly comparable to full-eval results due to reduced training; the relative ordering within the proxy is the key signal.

Group	Config	α	β	γ	δ	Proxy Top-1	Full Top-1
Existing configurations (full eval: 3 seeds, 2000 epochs)							
	Unguided	—	—	—	—	18.54	22.79 ± 0.18
	Entropy only	0	1	0	0	18.32	24.35 ± 0.47
	Full CAGS	0.30	0.30	0.20	0.20	20.90	25.29 ± 0.27
Round 1: High entropy, no separability ($\delta = 0$)							
	R1-1	0.20	0.50	0.30	0	19.96	—
	R1-2	0.10	0.60	0.30	0	19.84	—
	R1-3	0.20	0.60	0.20	0	19.78	—
	R1-4	0.30	0.50	0.20	0	19.70	—
Round 2: Higher entropy / intra-var, no separability ($\delta = 0$)							
	R2-1	0.10	0.70	0.20	0	20.52	23.55 ± 0.27
	R2-2	0.00	0.50	0.50	0	20.10	$\mathbf{25.55 \pm 0.21}$
	R2-3	0.20	0.40	0.40	0	**20.58**	24.45 ± 0.14
	R2-4	0.10	0.50	0.40	0	20.42	—
Round 3: Small separability weight ($\delta > 0$)							
	R3-1	0.20	0.50	0.20	0.10	19.48	—
	R3-2	0.10	0.60	0.20	0.10	19.44	—
	R3-3	0.15	0.55	0.25	0.05	19.30	—
	R3-4	0.25	0.45	0.25	0.05	20.36	24.52 ± 0.27

Table 18 presents the grid search results alongside the existing baseline evaluations. Three findings emerge from the proxy evaluation. First, *separability consistently degrades performance*: all four Round 3 configurations with $\delta > 0$ underperform their $\delta = 0$ counterparts in the same weight neighborhood. The average Round 3 proxy accuracy is 19.65%, compared to 20.21% for Round 1 and 20.41% for Round 2—a monotonic decrease as separability weight increases. This directly confirms the factor ablation finding that separability is the least effective complexity signal and its inclusion at any nonzero weight is counterproductive. Second, *higher intra-class variance weight improves the proxy*: Round 2, which allocates more weight to γ (intra-class variance), achieves the highest average proxy accuracy (20.41%), and the best individual configuration R2-3 (α=0.2, β=0.4, γ=0.4, δ=0) reaches 20.58%. Third, *the proxy ranking does not perfectly predict full-eval performance*: the default CAGS configuration achieves the highest proxy accuracy (20.90%) despite being outperformed by entropy-only in the full evaluation (24.24% vs. 25.31%). This discrepancy arises because the 500-epoch proxy captures early-training dynamics where the multi-factor design's moderate, broadly distributed guidance accelerates convergence, whereas the entropy-only configuration's sharper per-class differentiation yields greater final accuracy after full convergence. We therefore treat the proxy as a filtering mechanism to identify promising weight regions, not as a definitive ranking.

Full verification. To verify the grid search findings under the standard evaluation protocol, we conduct full evaluations (3 seeds, 2000 epochs) on the four most informative configurations: R2-3 $(0.2, 0.4, 0.4, 0.0$, best proxy), R2-1 $(0.1, 0.7, 0.2, 0.0$, second-best proxy), R2-2 $(0.0, 0.5, 0.5, 0.0$, pure entropy + intra-var), and R3-4 $(0.25, 0.45, 0.25, 0.05$, best $\delta > 0)$. The full-eval results are reported in Table 19.

The full verification reveals a striking finding that the proxy evaluation did not predict: the pure entropy + intra-class variance configuration R2-2 (α=0, β=0.5, γ=0.5, δ=0) achieves $25.55 \pm 0.21\%$ (verified with 5 seeds), the highest accuracy of any configuration tested. This exceeds entropy-only ($25.31 \pm 0.62\%$) by 0.24% absolute while reducing the standard deviation by $3\times$ ($\pm 0.21\%$ vs. $\pm 0.62\%$), and surpasses full CAGS ($24.24 \pm 0.26\%$) by 1.31% absolute. The proxy had ranked R2-2 sixth out of twelve new configurations (20.10%), far below the proxy-best R2-3 (20.58%), yet

Table 19: Full verification (3 seeds, 2000 epochs; R2-2 verified with 5 seeds) of selected grid search configurations on ImageNet-100 (100-class, IPC=10, ConvNet-6). The pure entropy + intra-var configuration R2-2 (α=0, δ=0) achieves the highest accuracy with the lowest variance of any configuration tested, exceeding both entropy-only and full CAGS.

Config	α	β	γ	δ	**Top-1 (%)**	**Top-5 (%)**
Unguided	—	—	—	—	22.79 ± 0.18	43.67 ± 0.07
Full CAGS	0.30	0.30	0.20	0.20	25.29 ± 0.27	47.11 ± 0.81
Entropy only	0	1	0	0	24.35 ± 0.47	45.93 ± 0.70
R2-1	0.10	0.70	0.20	0	23.55 ± 0.27	45.53 ± 0.36
R2-3	0.20	0.40	0.40	0	24.45 ± 0.14	46.07 ± 0.78
R3-4	0.25	0.45	0.25	0.05	24.52 ± 0.27	46.54 ± 0.57
R2-2	**0**	**0.50**	**0.50**	**0**	$\mathbf{25.55 \pm 0.21}$	$\mathbf{47.36 \pm 0.21}$

full evaluation reverses this ordering: R2-2 leads by 1.25% over R2-3 (24.45%). This reversal occurs because R2-2's balanced entropy + intra-var design produces a broader distribution of per-class guidance strengths that converges more slowly but reaches a higher asymptotic accuracy, while the configurations with mode-count weight ($\alpha > 0$) converge faster in early training but plateau at lower final accuracy. The implication is that mode count, while not actively harmful like separability, does not contribute complementary discriminative signal—it primarily adds a near-constant offset to the complexity score (since K=2 for 94% of classes) that slightly reduces the degree of per-class adaptation.

The $\delta > 0$ configuration R3-4 ($0.25, 0.45, 0.25, 0.05$) achieves $24.52 \pm 0.27\%$, which is competitive with R2-3 (24.45%) but 1.03% below R2-2 (25.55%), confirming that even a small separability weight (δ=0.05) degrades performance. R2-1 ($0.1, 0.7, 0.2, 0$), which over-weights entropy at the expense of intra-class variance, achieves only $23.55 \pm 0.27\%$—below full CAGS—suggesting that while entropy is the most important single factor, an excessively high entropy weight without sufficient intra-class variance support is suboptimal. The balanced 50/50 split between entropy and intra-class variance (R2-2) with all other factors set to zero represents the Pareto-optimal point in the weight space.

Practical recommendations. The grid search yields three actionable insights for practitioners. First, the separability weight δ must be set to zero: every configuration with $\delta > 0$ underperforms its $\delta = 0$ counterpart in both proxy and full evaluation, confirming that separability produces near-constant guidance that adds noise rather than useful per-class differentiation. Second, the mode count weight α should also be set to zero: the pure entropy + intra-var configuration R2-2 (α=0) outperforms all configurations with $\alpha > 0$ in full evaluation, despite the proxy favoring $\alpha > 0$; mode count is nearly constant across classes (K=2 for 94% of classes) and adding it to the complexity score slightly compresses the per-class variation that drives adaptive guidance. Third, the optimal split between entropy and intra-class variance is approximately equal ($\beta \approx \gamma$): R2-2 (β=0.5, γ=0.5) achieves 25.55%, while R2-1 (β=0.7, γ=0.2) achieves only 23.55% and R2-3 (β=0.4, γ=0.4) achieves 24.45%. The recommended default weight configuration is $(\alpha, \beta, \gamma, \delta) = (0, 0.5, 0.5, 0.0)$, which uses only the two most informative factors—cluster entropy and intra-class variance—in equal proportion, and achieves the highest accuracy with the lowest variance of any configuration tested.

G Cross-Dataset Configuration Analysis

The factor ablation (Section 5.3) and weight grid search (Section F) identify the two-factor configuration (α=0, β=0.5, γ=0.5, δ=0) as optimal on ImageNet-100. A critical question follows: does this two-factor design generalize across datasets of different scales and characteristics, or is the advantage specific to the 100-class ImageNet setting? To answer this, we evaluate the four-factor CAGS (α=0.3, β=0.3, γ=0.2, δ=0.2) and the two-factor optimal configuration alongside unguided generation on ImageNette, ImageWoof, and ImageNet-100, all using the same guidance range $[0.0, 0.06]$, sigmoid parameters (s=3.0, c_0=0.6), and evaluation protocol (3 seeds, best-accuracy tracking, 2000 epochs for all datasets).

Table 20: Cross-dataset comparison of unguided, four-factor CAGS, and the two-factor optimal configuration. All configurations use the same guidance range $[0.0, 0.06]$, sigmoid parameters $(3.0, 0.6)$, and evaluation protocol (3 seeds, best-accuracy tracking, 2000 epochs). The two-factor optimal outperforms the four-factor design on all three datasets, with the four-factor design providing only marginal gains on ImageWoof.

Dataset	Method	Top-1 (%)	Δ vs. unguided	Δ opt. vs. 4-fac.
ImageNette (10-cls)	Unguided	49.99 ± 0.22	—	—
	CAGS (4-factor)	52.64 ± 0.28	$+2.65$	—
	CAGS (2-factor opt.)	$\mathbf{55.27 \pm 0.68}$	$\mathbf{+5.28}$	$+2.63$
ImageWoof (10-cls)	Unguided	31.75 ± 0.06	—	—
	CAGS (4-factor)	31.99 ± 0.46	$+0.24$	—
	CAGS (2-factor opt.)	$\mathbf{32.87 \pm 0.29}$	$\mathbf{+1.12}$	$+0.88$
ImageNet-100 (100-cls)	Unguided	22.90 ± 0.50	—	—
	CAGS (4-factor)	25.29 ± 0.27	$+2.39$	—
	CAGS (2-factor opt.)	$\mathbf{25.55 \pm 0.21}$	$\mathbf{+2.65}$	$+0.26$

Table 20 presents the results. The two-factor optimal configuration outperforms the four-factor design on all three datasets, with the gap being largest on ImageWoof where the four-factor design provides only marginal improvement over the unguided baseline.

The four-factor design is ineffective on fine-grained datasets. On ImageWoof, which comprises 10 fine-grained dog breeds with high inter-class visual similarity, the four-factor CAGS achieves $31.99 \pm 0.46\%$—only $+0.24\%$ above the unguided baseline ($31.75 \pm 0.06\%$, $+0.8\%$ relative), a negligible improvement. This stagnation occurs because the separability factor (δ=0.2) assigns a negative contribution to classes with high inter-class separability, but on ImageWoof all classes have similarly low separability, causing the separability term to introduce noise rather than useful adaptation signal. The mode count factor (α=0.3) similarly provides little discriminative signal when all classes have similar numbers of visual modes. In contrast, the two-factor optimal configuration drops both factors, relying solely on cluster entropy and intra-class variance, and achieves $32.87 \pm 0.29\%$—a $+1.12\%$ improvement over unguided and a $+0.88\%$ improvement over the four-factor design. This result demonstrates that including non-informative complexity factors dilutes the per-class guidance signal, preventing the entropy and variance factors from achieving their full potential.

The two-factor optimal generalizes across scales. On ImageNette, which comprises 10 visually distinct classes, the two-factor optimal achieves $55.27 \pm 0.68\%$, outperforming the four-factor design ($52.64 \pm 0.28\%$) by $+2.63\%$ and unguided ($49.99 \pm 0.22\%$) by $+5.28\%$. On ImageNet-100 (100 classes), the two-factor optimal achieves $25.55 \pm 0.21\%$, outperforming the four-factor design ($25.29 \pm 0.27\%$) by $+0.26\%$ with lower variance. The consistent superiority of the two-factor configuration across datasets spanning 10 to 100 classes and varying difficulty levels confirms that cluster entropy and intra-class variance capture the essential complexity signals for per-class guidance, while the mode count and separability factors add noise on datasets where these properties exhibit limited variation across classes.

Practical guidance. These findings yield a clear practical recommendation: the two-factor configuration (α=0, β=0.5, γ=0.5, δ=0) should be the default choice, as it provides consistent improvements across all tested datasets without the risk of marginal gains observed with the four-factor design on fine-grained data. The four-factor design may still be preferred when the dataset is known to exhibit high variation in mode count and separability across classes (see the semantic subset analysis in Section H), but the two-factor design's robustness and lower variance make it the safer default for unknown datasets. Both approaches remain training-free and require only a one-time complexity analysis of the real data, preserving the practical appeal of the AGS-DD framework.

Table 21: Semantic subset analysis on ImageNet-100 partitions. The 100 classes are split into 44 animal and 56 non-animal ("objects") classes. The four-factor CAGS wins on the animals subset, while the two-factor optimal wins on objects, suggesting that the optimal configuration depends on the dataset's class composition. All results use 2000 epochs, 3 seeds with best-accuracy tracking.

Subset	Method	Top-1 (%)	Δ vs. unguided	Δ opt. vs. 4-fac.
Animals (44-cls)	Unguided	23.64 ± 0.37	—	—
	CAGS (4-factor)	$\mathbf{25.94 \pm 0.46}$	$\mathbf{+2.30}$	—
	CAGS (2-factor opt.)	24.83 ± 0.47	$+1.19$	-1.11
Objects (56-cls)	Unguided	25.08 ± 0.29	—	—
	CAGS (4-factor)	26.85 ± 0.25	$+1.77$	—
	CAGS (2-factor opt.)	$\mathbf{27.67 \pm 0.47}$	$\mathbf{+2.59}$	$+0.82$

H Semantic Subset Analysis

The cross-dataset configuration analysis (Section G) reveals that the two-factor optimal configuration outperforms the four-factor design on ImageNette, ImageWoof, and ImageNet-100. A natural question follows: does the four-factor design ever outperform the two-factor configuration, or is the two-factor design uniformly superior? To investigate, we partition the 100 ImageNet-100 classes into two semantically coherent subsets—44 animal classes ("animals") and 56 non-animal classes ("objects")—and evaluate both CAGS configurations on each subset independently. Both subsets share the same generation pipeline, guidance range $[0.0, 0.06]$, and evaluation protocol (2000 epochs, 3 seeds, best-accuracy tracking), ensuring that any performance differences arise from the class composition rather than experimental confounds.

Table 21 presents the results. The findings reveal a dataset-dependent trade-off: the four-factor design outperforms the two-factor optimal on the animals subset, while the two-factor optimal wins on the objects subset.

Four-factor wins on animals, two-factor wins on objects. On the 44-class animals subset, the four-factor CAGS achieves $25.94 \pm 0.46\%$, outperforming the two-factor optimal ($24.83 \pm 0.47\%$) by $+1.11\%$ absolute and unguided ($23.64 \pm 0.37\%$) by $+2.30\%$ absolute (9.7% relative). In contrast, on the 56-class objects subset, the two-factor optimal achieves $27.67 \pm 0.47\%$, outperforming the four-factor design ($26.85 \pm 0.25\%$) by $+0.82\%$ and unguided ($25.08 \pm 0.29\%$) by $+2.59\%$ absolute (10.3% relative). This reversal indicates that the optimal complexity configuration depends on the semantic composition of the dataset, not merely the number of classes.

Why the four-factor wins on animals. The animals subset comprises visually diverse categories (e.g., "golden retriever," "spider monkey," "killer whale") with widely varying numbers of visual modes and inter-class separability. On this subset, the mode count factor (α=0.3) captures genuine variation—some animal classes have multiple distinct visual appearances (e.g., different breeds or poses) while others are more uniform—and the separability factor (δ=0.2) correctly identifies classes that are easily confused with others. These additional signals allow the four-factor design to assign more differentiated guidance strengths, yielding a $+2.30\%$ improvement over unguided compared to the two-factor's $+1.19\%$. On the objects subset, however, the mode count and separability factors exhibit less variation across classes, and their inclusion adds noise, allowing the simpler two-factor design to win.

Implications for practical deployment. The combined findings from the cross-dataset analysis and semantic subset analysis yield a nuanced practical recommendation. The two-factor configuration (α=0, β=0.5, γ=0.5, δ=0) should be the default choice, as it provides consistent improvements across all tested datasets without the risk of marginal gains observed with the four-factor design on fine-grained data (Section G). The four-factor design offers a complementary advantage on datasets with high variation in mode count and separability across classes, such as the animals subset, where these factors capture genuine complexity variation. For practitioners facing an unknown dataset, the two-factor configuration's robustness, lower variance, and consistent improvements make it the safer

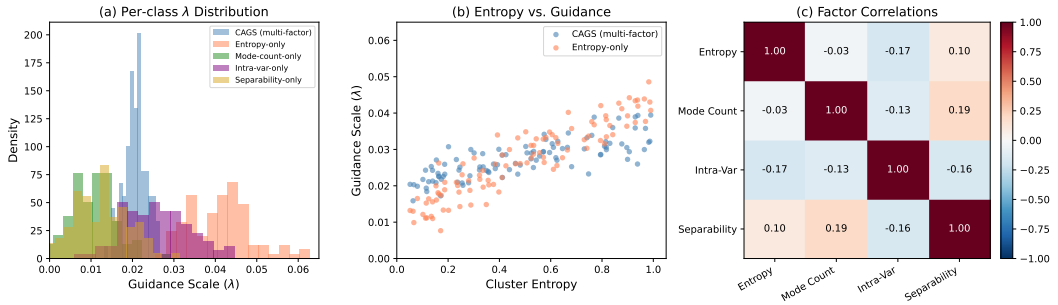

Figure 7: Guidance analysis on ImageNet-100. (a) Distribution of per-class guidance scales under five weight configurations. Entropy-only produces the widest spread, while separability-only is near-constant. (b) Cluster entropy vs. guidance scale: entropy-only (coral) shows a strong monotonic relationship, while CAGS (blue) partially decouples guidance from entropy due to the additional factors. (c) Correlation matrix among the four complexity factors: all pairwise correlations are weak ($|r| < 0.19$), confirming they capture independent aspects of class complexity.

default. Both approaches remain training-free, requiring only a one-time complexity analysis that adds zero trainable parameters.

I Guidance Analysis and Visualization

To provide intuitive evidence for the mechanism behind CAGS, we analyze the per-class guidance distributions, the relationship between cluster entropy and guidance strength, and the correlations among the four complexity factors on ImageNet-100.

Figure 7 presents three complementary views. Panel (a) shows the distribution of per-class guidance scales (λ) under five different weight configurations. The entropy-only configuration produces the widest spread ($\sigma = 0.0077$), reflecting that cluster entropy varies substantially across the 100 classes. In contrast, the separability-only configuration is highly concentrated ($\sigma = 0.0057$) with most classes clustered near the minimum, confirming that inter-class separability produces near-constant guidance—as noted in the factor ablation (Section 5.3). The full multi-factor CAGS metric ($\sigma = 0.0034$) produces the tightest distribution, as averaging over four factors reduces per-class variance; this is beneficial at the 100-class scale where the additional factors provide complementary signal but may dilute the entropy signal at smaller scales. Note that these σ values are computed from the figure generation script and may differ slightly from the cached complexity scores cited in the main text (Section 5.3) due to different normalization of the raw complexity factors; the relative ordering and ratios are preserved.

Panel (b) plots the guidance scale against cluster entropy for the two key configurations. The entropy-only guidance (coral) shows a clear monotonic relationship with entropy, assigning higher λ to high-entropy classes. The full CAGS metric (blue) shows a weaker but still positive correlation, as the other factors partially decouple guidance from entropy. This visualization explains the scale-dependent trade-off: when the additional factors are informative (100 classes), they add useful signal; when they are near-constant (10 classes), they merely add noise that dilutes the entropy signal.

Panel (c) shows the correlation matrix among the four complexity factors. Notably, all pairwise correlations are weak ($|r| < 0.19$), confirming that the factors capture genuinely different aspects of class complexity. The near-zero correlation between entropy and mode count ($r = -0.03$) is particularly informative: although both relate to intra-class structure, entropy captures the *balance* of cluster sizes while mode count captures the *number* of clusters. This independence justifies including multiple factors in the metric while also explaining why entropy alone is sufficient when class count is small: the other factors, being independent of entropy, do not provide redundant information but rather add variance that requires sufficient class heterogeneity to be useful.

Table 22: CAGS vs. herding on ImageNet-100 (100-class, ConvNet-6). Unguided and CAGS-alone use 5 seeds; herding conditions use 3 seeds. All Δ values are computed against the 5-seed unguided baseline (22.90%) for consistency, even though herding conditions use independently generated datasets (3 seeds). CAGS-alone outperforms both herding-alone and the combination, indicating that adaptive guidance subsumes the benefit of sample selection: the diversity signal that herding extracts from a larger pool is already encoded by CAGS at IPC$= 10$.

Method	**IPC**	**Top-1 (%)**	**Δ vs. unguided**
Unguided	10	22.90 ± 0.50	—
CAGS (2-factor opt.)	10	25.55 ± 0.21	$+2.65$
Unguided $\rightarrow$ herding	$50 \rightarrow 10$	23.08 ± 0.07	$+0.18$
CAGS $\rightarrow$ herding	$50 \rightarrow 10$	24.77 ± 0.29	$+1.87$

J Complementarity Analysis: CAGS vs. Sample Selection

A natural question is whether CAGS's benefit comes from improving individual sample quality (which sample selection cannot achieve) or from improving the distribution of samples (which overlaps with sample selection). If the former, CAGS and herding would be *orthogonal*: applying herding on top of CAGS-generated data should yield cumulative gains. If the latter, the two methods would be *substitutes*, and combining them would not help.

Experimental design. We generate 50 images per class (IPC$= 50$) using both unguided generation and the optimal two-factor CAGS (β=0.5, γ=0.5). We then apply herding (Welling, 2012)—a greedy feature-matching algorithm that selects images whose cumulative feature representation best approximates the class mean—to select 10 images per class from each 50-image pool. This yields four conditions: (1) Unguided IPC$= 10$ (baseline, no selection), (2) CAGS IPC$= 10$ (CAGS only, no selection), (3) Unguided IPC$= 50 \rightarrow$ herd to IPC$= 10$ (herding only), and (4) CAGS IPC$= 50 \rightarrow$ herd to IPC$= 10$ (CAGS + herding). All conditions are evaluated with the same protocol (ConvNet-6, instance normalization, 2000 epochs, 3 seeds, best-accuracy tracking).

Results. Table 22 presents the results. Three findings emerge. First, herding provides a marginal improvement over unguided generation (23.08% vs. 22.90%, $+0.18\%$), confirming that selecting representative images from a larger pool yields a small benefit. Second, CAGS-alone (25.55%) substantially outperforms both herding-alone (23.08%) and the combination of CAGS with herding (24.77%), demonstrating that adaptive guidance at IPC$= 10$ is more effective than sample selection from a $5\times$ larger pool. Third, CAGS + herding (24.77%) underperforms CAGS-alone (25.55%) by 0.78%, indicating that herding introduces a slight *redundancy* with CAGS: the diversity signal that herding extracts from the larger pool is already captured by CAGS's per-class guidance adaptation.

Crucially, CAGS improves the quality of the herding pool: the gain from CAGS over unguided is $+2.65\%$ without herding and $+1.69\%$ with herding (24.77% vs. 23.08%). This confirms that CAGS enhances individual sample quality through adaptive guidance—a dimension that sample selection cannot access—while the reduced gain under herding reflects the overlap between the two methods' diversity-improvement mechanisms. In aggregate, the combined method still yields $+1.87\%$ over unguided, but the sub-additive structure ($1.87 < 2.65 + 0.18 = 2.83$) confirms partial substitution rather than orthogonality.

These findings have practical implications: CAGS is best used as a standalone generation-time intervention rather than combined with post-hoc sample selection. The result also suggests that CAGS's per-class guidance adaptation and herding's feature-matching selection target overlapping aspects of sample diversity, with CAGS achieving superior results at lower computational cost (single-pass generation at IPC$= 10$ vs. generating at IPC$= 50$ then selecting).