DMGD: Train-Free Dataset Distillation with Semantic-Distribution Matching in Diffusion Models

Supplementary Material

A1: Background

A1.1: Diffusion Sample process

Diffusion models [26, 40, 59] comprise a forward process $\{q(\boldsymbol{x}_t)\}_{t\in[0,T]}$ that gradually adds noise to data $\boldsymbol{x}_0 \sim q(\boldsymbol{x}_0)$, alongside a learned reverse process $\{p(\boldsymbol{x}_t)\}_{t\in[0,T]}$ targeting to denoise the data.

The forward process is formulated as $q(\boldsymbol{x}_t|\boldsymbol{x}_0) := \mathcal{N}(\sqrt{\alpha_t}\boldsymbol{x}_0, (1-\alpha_t)\mathbf{I})$ and $q(\boldsymbol{x}_t) := \int q(\boldsymbol{x}_t|\boldsymbol{x}_0)q(\boldsymbol{x}_0)\mathrm{d}\boldsymbol{x}_0$, with α_t representing a noise schedule. The reverse process, initialized from $p(\boldsymbol{x}_T) := \mathcal{N}(\mathbf{0}, \mathbf{I})$, is characterized by a parameterized denoiser $\boldsymbol{\epsilon}_\theta^t(\boldsymbol{x}_t)$, which aims to predict the noise added to $\boldsymbol{x}_0$. The denoiser $\epsilon_\theta$can be optimized by minimizing:

$$\mathcal{L}_{\mathrm{DM}} := \mathbb{E}_{x_0,t,\boldsymbol{\epsilon}}[w(t)\left\|\boldsymbol{\epsilon}_\theta^t(\sqrt{\alpha_t}\boldsymbol{x}_0 + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}) - \boldsymbol{\epsilon}\right\|_2^2] \tag{1}$$

where $\boldsymbol{x}_0 \sim q(\boldsymbol{x}_0), t \sim \mathcal{U}(0,T), \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $w(t)$ is a pre-specified weight function. A more widely adopted approach is the Latent Diffusion Model (LDM) [5, 49, 79], which leverages a Variational Autoencoder (VAE) [29] to compress input x into latent space samples z, followed by executing diffusion within this latent space. In this work, we employ LDM as a pretrained backbone model requiring no additional training, and adopt the sampling process defined by DDIM (Denoising Diffusion Implicit Models) [39, 59]. DDIM first maps the noisy sample z_t back to the clean data distribution, obtaining $z_{0|t}$. Then, it samples z_{t-1} through the diffusion process:

$$z_{0|t} = \frac{z_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(z_t, t, c)}{\sqrt{\bar{\alpha}_t}} \tag{2}$$

We can finally obtain the single-step denoising result via the DDIM sampling formula:

$$z_{t-1} = \alpha_t^1 z_{0|t}(z_t) + \alpha_t^2 \epsilon_\theta(z_t, t, c) + \alpha_t^3 \epsilon \tag{3}$$

where $\alpha_t^1 = \sqrt{\bar{\alpha}_{t-1}}$, $\alpha_t^2 = \sqrt{1-\bar{\alpha}_{t-1}-\eta^2(1-\bar{\alpha}_t)}$, $\alpha_t^3 = \eta\sqrt{1-\bar{\alpha}_{t-1}}$. η is predefined noise factor. For compact representation, we define whole process as $z_{t-1} = DDIM(z_t, t, c)$. Furthermore, we can incorporate other conditional gradient guidance during sampling to achieve guided diffusion [76]. Given a differentiable conditioning function $E(z_t, c)$, where c represents a conditional input of arbitrary form, we can define a single-step guided diffusion process as:

$$z_{t-1} = DDIM(z_t) - \rho_t \nabla E(z_t, c) \tag{4}$$

However, directly evaluating $E(z_t, c)$ on noisy samples z_t is challenging. Thus, we approximate it by computing it at the mapped point of z_t on the clean data manifold, i.e., $E(z_t, c) \approx \hat{E}(z_{0|1}(z_t), c)$, where $z_{0|1}$ is the denoised estimate of z_t via Eq. (2).

Entropy-Regularized OT and Sinkhorn Algorithm To address the computational challenges of OT, *entropy regularization* introduces a penalization term to the objective, smoothing the transport plan γ and enabling efficient computation. The entropy-regularized OT problem is defined as:

$$W_\varepsilon(\mathbf{a}, \mathbf{b}) = \min_{\gamma\in\Pi(\mathbf{a},\mathbf{b})} \langle\gamma, \mathbf{C}\rangle - \varepsilon H(\gamma), \tag{5}$$

where $\varepsilon > 0$ controls the strength of the regularization, and $H(\gamma) = -\sum_{i,j}\gamma_{ij}\log\gamma_{ij}$ is the entropy of the transport plan. The regularization makes the problem strictly convex and allows for efficient iterative solutions, even for large $\mathbf{a}, \mathbf{b}$. The Sinkhorn algorithm is an iterative method to solve the entropy-regularized OT problem. It leverages the fact that the optimal transport plan γ^* under entropy regularization can be expressed in a factorized form:

$$P_{ij}^* = \mathbf{a}_i\mathbf{b}_j \exp\left(-\frac{C_{ij}}{\epsilon} + u_i + v_j\right) \tag{6}$$

where $u \in \mathbb{R}^n$ and $v \in \mathbb{R}^m$ are dual variables ensuring the marginal constraints are satisfied. Rearranging, this simplifies to:

$$P^* = \mathrm{diag}(u)\, K\, \mathrm{diag}(v) \tag{7}$$

where $K \in \mathbb{R}_+^{n\times m}$ is the kernel matrix defined as $K_{ij} = \exp\left(-\frac{C_{ij}}{\epsilon}\right)$, and $\mathrm{diag}(u)$ (resp. $\mathrm{diag}(v)$) is a diagonal matrix with u (resp. v) on the diagonal. In practice, the Sinkhorn algorithm alternates between updating u and v to enforce the marginal constraints. Starting with initial guesses $u_0 = \mathbf{1}$ (all ones) and $v_0 = \mathbf{1}$, the updates are:

$$\begin{aligned} v_k &= \frac{\beta}{K^\top u_{k-1}}, \\ u_k &= \frac{\alpha}{K v_k} \end{aligned} \tag{8}$$

where $K^\top$ denotes the transpose of K, and division is element-wise. After T iterations, the transport plan is approximated as $P \approx \mathrm{diag}(u_T)K\mathrm{diag}(v_T)$. Besides, A simplified and numerically stable variant of the Sinkhorn algorithm is the *row-column normalization method*, which directly operates on the kernel matrix K without explicitly

tracking u and v. The key insight is that alternating row and column normalization of K enforces the marginal constraints α and β iteratively [14]. The steps are as follows:

$$K_{\text{row}} = K \odot \left(\frac{\alpha}{\text{row\_sum}(K)} \right) \tag{9}$$

$$K = K_{\text{row}} \odot \left(\frac{\beta}{\text{col\_sum}(K_{\text{row}})} \right) \tag{10}$$

where $\text{row\_sum}(K) \in \mathbb{R}^n$ is the vector of row sums of K, and $\odot$ denotes element-wise multiplication. $\text{col\_sum}(K_{\text{row}}) \in \mathbb{R}^m$ is the vector of column sums of K_{row}. After T iterations, the normalized K itself serves as the approximate transport plan $P \approx K$.

A1.2: More Related Work

Optimization Based Methods. Optimization based methods are classical dataset distillation algorithms. They align representations or training dynamics between synthetic datasets ($\mathcal{S}$) and real datasets ($\mathcal{T}$) via matching losses, and update the synthetic dataset through gradient optimization. Gradient Matching (GM), one of the earliest dataset distillation algorithms, updates samples by matching training gradients on $\mathcal{S}$ and $\mathcal{T}$ [57, 80, 81]. However, GM requires simultaneous gradient updates for both samples and the model, leading to a bi-level optimization dilemma. In contrast, Trajectory Matching (TM) aims to directly match training trajectories between $\mathcal{S}$ and $\mathcal{T}$ without complex gradient computations [6, 13, 18, 23, 34, 83]. Guo et al. [23] observed that different parameter trajectories can be adopted for distillation across datasets, achieving lossless distillation on small-scale datasets for the first time. Distribution Matching (DM) seeks to ensure that $\mathcal{S}$ effectively covers $\mathcal{T}$ in the feature space, i.e., matching their feature distributions [37, 52, 54, 66, 82]. Zhao et al. [82] proposed using randomly initialized feature extractors for mapping and matching the means of $\mathcal{T}$ and $\mathcal{S}$ to approximate distribution matching. [37] proposed selecting representative data via K-means clustering for matching. Optimal Transport is regarded as a key insight for enhancing distribution matching. [35] proposed using the Wasserstein barycenter of the $\mathcal{T}$ as matching targets. OPTICAL [14] leverages mini-batch optimal transport to improve the matching relationship between samples in $\mathcal{S}$ and $\mathcal{T}$. Our method also draws on the key insight of optimal transport, designing a new OT-guided loss for the diffusion based dataset distillation framework. We further propose two key strategies: approximate distribution matching and greedy progressive matching, to ensure performance while further optimizing efficiency.

Disentangled Dataset Distillation Disentangled dataset distillation frameworks have successfully overcome the bi-level optimization dilemma, extending dataset distillation to large-scale datasets such as ImageNet[28, 36, 52, 56, 61, 70, 72, 75]. SRe2L [75] proposed a squeeze-recover-relabel paradigm: first, it squeezes the key information of the dataset into a neural network through training; then, it optimizes samples through designed matching losses for recovery; finally, it performs relabeling based on the pretrained model. G-VBSM [52] extended such methods via large-scale statistical matching and multi-backbone model. Xiao and He [70] proposed a label pruning method to optimize the label space, significantly reducing the storage space of such methods. Xue et al. [72] proposed a curvature regularization loss to improve the adversarial robustness of disentangled dataset distillation. Inspired by these approaches, Sun et al. [61] introduced a non-optimization framework RDED, which conducts dataset distillation by directly extracting effective patches using a pre-trained model. Inspired by this category of methods, we designed semantic matching and distribution matching objectives for diffusion based dataset distillation. Meanwhile, we further improved the matching framework specifically for diffusion models.

Generative Model Based Dataset Distillation In contrast to methods based on discriminative models, generative model based approaches can synthesize data that exhibits high consistency with the original dataset. This consistency (also termed realism) effectively enhances cross-architecture performance. Prior research [7, 65, 77, 84] proposed using Generative Adversarial Networks (GANs) [20] as prior models for dataset distillation, synthesizing realistic data by optimizing latent space variables. [78] extended GAN-based dataset distillation methods to the image super-resolution setting, further validating the immense potential of generative models in dataset distillation. Recently, researchers have increasingly focused on applying diffusion models [12, 26, 59] to dataset distillation [8, 10, 22, 60]. Minimax [22] introduced an efficient fine-tuning-based method [71] to further align diffusion models with target datasets. D^4M [60] proposed a disentangled diffusion model framework: it first extracts mode means via K-means and generates representative samples through DDIM inversion; subsequently, it employs knowledge distillation for soft label annotation. MGD^3 [8] devised a training-free guided diffusion model framework for dataset distillation, comprising three stages: mode discovery, mode guidance, and stop guidance. However, this method lacks attention to the distribution structure, which may lead to overemphasizing invalid mode points. IGD [10] introduce trajectory matching into diffusion model guidance, utilizing a auxiliary trained classifier to steer generation toward high-influence samples. However, complex trajectory optimization causes it to lose the efficient characteristics of diffusion based methods. Independently and concurrently with our work, [15] explored the application of optimal transport-

based diffusion models in dataset distillation, with a specific focus on how optimal transport relates to soft label learning within this task. Our method rethinks the framework for applying diffusion models to dataset distillation, proposing two core objectives: semantic matching and distribution matching. For semantic matching, we demonstrate that diffusion models effectively inject semantic information and design a dynamic soft labeling approach to enhance diversity. For distribution matching, we propose an optimal transport-guided loss that effectively aligns the distribution of generated samples with the real dataset without requiring additional model training.

A2: Proof

In this section, we will provide detailed proofs for the theoretical analyses presented in the paper, and in conjunction with the design space of dataset distillation, discuss how these theories guide the design of our DMGD framework.

A2.1: Proof of Theorem 1

Theorem 1 *Let $\mathcal{T}$ and $\mathcal{S}$ denote the target and surrogate datasets, respectively, with $\theta^*_{\mathcal{T}}$ and $\theta^*_{\mathcal{S}}$ being their optimally trained parameters. Define the target risk as: $R_{\mathcal{T}}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{T}}\left[\ell(x,y,\theta)\right]$, where $\ell(\cdot)$ is an L-Lipschitz continuous evaluation function. Under semantic class alignment (i.e., no label mismatch), consider the marginal sample distributions $P_{\mathcal{T}}$ and $P_{\mathcal{S}}$ with optimal transport distance: $W(P_{\mathcal{T}}, P_{\mathcal{S}}) = \inf_{\gamma\in\Gamma(P_{\mathcal{T}},P_{\mathcal{S}})}\mathbb{E}_{(x_{\mathcal{T}},x_{\mathcal{S}})\sim\gamma}\left[d(x_{\mathcal{T}},x_{\mathcal{S}})\right]$, where $\Gamma(P_{\mathcal{T}}, P_{\mathcal{S}})$ is the set of all couplings between the distributions, and $d(\cdot,\cdot)$ is a metric on the sample space. Then the risk discrepancy satisfies:*

$$|R_{\mathcal{T}}(\theta^*_{\mathcal{T}}) - R_{\mathcal{T}}(\theta^*_{\mathcal{S}})| \leq 2L \cdot W(P_{\mathcal{T}}, P_{\mathcal{S}}). \tag{11}$$

Proof. Through the optimal properties of parameters θ^*, we decompose the risk discrepancy:

$$\begin{aligned}\Delta &= R_{\mathcal{T}}(\theta^*_{\mathcal{S}}) - R_{\mathcal{T}}(\theta^*_{\mathcal{T}}) \\ &= R_{\mathcal{T}}(\theta^*_{\mathcal{S}}) - R_{\mathcal{S}}(\theta^*_{\mathcal{S}}) + R_{\mathcal{S}}(\theta^*_{\mathcal{S}}) - R_{\mathcal{T}}(\theta^*_{\mathcal{T}}) \\ &\leq \underbrace{R_{\mathcal{T}}(\theta^*_{\mathcal{S}}) - R_{\mathcal{S}}(\theta^*_{\mathcal{S}})}_{I} + \underbrace{R_{\mathcal{S}}(\theta^*_{\mathcal{T}}) - R_{\mathcal{T}}(\theta^*_{\mathcal{T}})}_{II}\end{aligned} \tag{12}$$

For conciseness, we define $\Delta = |R_{\mathcal{T}}(\theta^*_{\mathcal{T}}) - R_{\mathcal{T}}(\theta^*_{\mathcal{S}})|$. We review the definition of risk $R_{\mathcal{T}}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{T}}\left[\ell(x,y,\theta)\right]$. Due to the consistency of labels and the consistency of parameters, we can express the first term as:

$$I = \mathbb{E}_{x\sim P_{\mathcal{T}}}\left[\ell_{\theta^*_s}(x)\right] - \mathbb{E}_{x\sim P_{\mathcal{S}}}\left[\ell_{\theta^*_s}(x)\right] \tag{13}$$

To explain the risk discrepancy from the perspective of optimal transport theory, we introduce the key lemma for Theorem 1: the Kantorovich-Rubinstein duality (**Lemma 2**).

Lemma 2 (Kantorovich-Rubinstein Duality [64]) *Let $(\mathcal{X}, d)$ be a complete separable metric space (Polish space). For any Borel probability measures $\mu, \nu \in \mathcal{P}_1(\mathcal{X})$ with finite first moments, the Wasserstein distance admits the dual representation:*

$$\begin{aligned}W(\mu,\nu) &= \inf_{\gamma\in\Gamma(\mu,\nu)}\mathbb{E}_{(x,y)\sim\gamma}[d(x,y)] \\ &= \sup_{f\in Lip_1(\mathcal{X})}\left(\mathbb{E}_{x\sim\mu}[f(x)] - \mathbb{E}_{y\sim\nu}[f(y)]\right)\end{aligned} \tag{14}$$

where, $\Gamma(\mu,\nu)$ denotes the set of couplings with marginals μ and ν. $\|f\|_{Lip} = \sup_{x\neq y}\frac{|f(x)-f(y)|}{d(x,y)}$ is the Lipschitz semi-norm. $\mathcal{P}_1(\mathcal{X})$ is the space of probability measures with $\int d(x_0,x)d\mu(x) < \infty$ for some $x_0 \in \mathcal{X}$

Let $\mu = P_{\mathcal{T}}$, $\nu = P_{\mathcal{S}}$. Meanwhile, since ℓ satisfies L-Lipschitz continuity, we can set $f(x) = \frac{\ell_{\theta^*_{\mathcal{S}}}(x)}{L}$. Building on the basis of Lemma 2, we have:

$$\begin{aligned}I &= \mathbb{E}_{x\sim P_{\mathcal{T}}}\left[\ell_{\theta^*_{\mathcal{S}}}(x)\right] - \mathbb{E}_{x\sim P_{\mathcal{S}}}\left[\ell_{\theta^*_{\mathcal{S}}}(x)\right] \\ &= L\cdot\left(\mathbb{E}_{x\sim P_{\mathcal{T}}}\left[f(x)\right] - \mathbb{E}_{x\sim P_{\mathcal{S}}}\left[f(x)\right]\right) \\ &\leq L\cdot\sup_{f\in \mathrm{Lip}_1(\mathcal{X})}\left(\mathbb{E}_{x\sim P_{\mathcal{T}}}[f(x)] - \mathbb{E}_{y\sim P_{\mathcal{S}}}[f(y)]\right) \\ &= L\cdot W(P_{\mathcal{T}}, P_{\mathcal{S}})\end{aligned} \tag{15}$$

Similarly, for the second term, we have:

$$\begin{aligned}II &= \mathbb{E}_{x\sim P_{\mathcal{S}}}\left[\ell_{\theta^*_{\mathcal{T}}}(x)\right] - \mathbb{E}_{x\sim P_{\mathcal{T}}}\left[\ell_{\theta^*_{\mathcal{T}}}(x)\right] \\ &= L\cdot\left(\mathbb{E}_{x\sim P_{\mathcal{S}}}\left[f(x)\right] - \mathbb{E}_{x\sim P_{\mathcal{T}}}\left[f(x)\right]\right) \\ &\leq L\cdot\sup_{f\in \mathrm{Lip}_1(\mathcal{X})}\left(\mathbb{E}_{x\sim P_{\mathcal{S}}}[f(x)] - \mathbb{E}_{y\sim P_{\mathcal{T}}}[f(y)]\right) \\ &= L\cdot W(P_{\mathcal{S}}, P_{\mathcal{T}})\end{aligned} \tag{16}$$

Combining the two terms, based on the symmetric property of the optimal transport distance, i.e. $W(P_{\mathcal{S}}, P_{\mathcal{T}}) = W(P_{\mathcal{T}}, P_{\mathcal{S}})$, we can derive Theorem 1:

$$|R_{\mathcal{T}}(\theta^*_{\mathcal{T}}) - R_{\mathcal{T}}(\theta^*_{\mathcal{S}})| \leq 2L \cdot W(P_{\mathcal{T}}, P_{\mathcal{S}}). \tag{17}$$

Discussion The core idea of **Theorem 1** is to decompose the objectives of dataset distillation into two domain adaptation objectives [11, 30, 62, 68] concerning the optimal parameters on the target dataset and the optimal parameters on the surrogate dataset, respectively. This decomposition bridges the gap between the fields of dataset distillation and domain adaptation. However, traditional domain adaptation algorithms optimize model parameters, while dataset distillation optimizes synthetic samples. This discrepancy in optimization objects makes it challenging to apply optimal transport to optimizing the joint distribution of samples and labels in the context of dataset distillation. Meanwhile, image data has the characteristic of redundancy in information dimensions, which means the semantic information only occupies a small number of dimensions in the pixel or feature

space. Performing optimal transport solely on the sample distribution fails to preserve representative semantic information.

Therefore, we aim to handle the alignment of semantic information and distribution structures separately, which also constitutes the starting point of our **Theorem 1** and the DMGD framework. **Theorem 1** indicates that, under certain constraint guidance such that semantic alignment is satisfied, optimizing the optimal transport distance between the surrogate dataset and the target dataset is equivalent to optimizing the upper bound of the risk discrepancy. Therefore, we only need to consider that surrogate samples must have semantic information consistent with the target class, i.e., semantic alignment. We can define semantic alignment from the perspective of conditional likelihood.

Definition 2 (Semantic Alignment) *Let $\mathcal{X}$ be the sample space, $\mathcal{Y} = \{y_1, \ldots, y_m\}$ a finite label set of semantic categories, and $\log p(\cdot|x)$ a conditional log-likelihood distribution over $\mathcal{Y}$ for a given sample $x \in \mathcal{X}$. A sample x and target semantic label $y \in \mathcal{Y}$ are* semantically aligned *if and only if:*

$$y = \arg\max_{y^* \in \mathcal{Y}} \log p(y^*|x) \tag{18}$$

By **Definition 2**, we can achieve semantic alignment by optimizing the conditional log-likelihood $\log p(y|x)$. In discriminative models, the conditional log-likelihood can be estimated from the softmax output of the classifier, and synthetic samples can be optimized via backpropagation [52, 70, 75]. In generative models, especially diffusion models, classifier-free guidance [22, 25, 60] is an effective method for estimating and optimizing conditional log-likelihood. This makes it feasible to align semantics within the diffusion model framework without the need for additional classifier training. This also forms the design basis of our semantic matching.

For distribution matching, we still need to first consider whether distribution alignment will lead to a mismatch of semantic information, which is also the premise for handling the two objectives separately. The traditional setup of dataset distillation provides a natural way to meet the assumptions by distilling instances for each class distribution. We perform distribution matching for each class separately to disentangle semantic information. Through optimal transport matching on class distributions, we can obtain the objective of distribution alignment that guides practice.

$$\arg\min_{\mathcal{S}^c} W(P_{\mathcal{S}^c}, P_{\mathcal{T}^c}) \tag{19}$$

$\mathcal{S}^c$ is the set of instances assigned to class c in the surrogate dataset, and $\mathcal{T}^c$ is the set of samples labeled c in the target dataset.

A2.2: Proof of Lemma 1

Lemma 1 (Classifier-Free Guidance [25]) *Consider a noise prediction network $\boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t, t, y)$, where $\boldsymbol{z}_t$ denotes the representation of an original sample $\boldsymbol{x}$ at timestep t, and y is a label. Assuming the $\boldsymbol{\epsilon}$ models both the conditional generative distribution $p(\boldsymbol{z}_t|y)$ and the unconditional distribution $p(\boldsymbol{z}_t)$, the gradient of the conditional log-likelihood $\log p(y|\boldsymbol{z}_t)$ with respect to $\boldsymbol{z}_t$ can be implicitly approximated by the difference between the network's conditional and unconditional outputs:*

$$\nabla_{\boldsymbol{z}_t} \log p(y|\boldsymbol{z}_t) \approx \omega \left(\boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t, t, \emptyset) - \boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t, t, y) \right) \tag{20}$$

Here, ω denotes a scalar guidance scale, and $\boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t, t, \emptyset)$ represents the network's unconditional output (i.e., without a specified class label).

Proof. By Bayes' theorem, the conditional likelihood decomposes as:

$$p(y|z_t) = \frac{p(z_t|y) \cdot p(y)}{p(z_t)} \tag{21}$$

Taking the logarithm and differentiating with respect to z_t:

$$\nabla_{z_t} \log p(y|z_t) = \nabla_{z_t} \log p(z_t|y) - \nabla_{z_t} \log p(z_t) \tag{22}$$

In diffusion models, the score functions relate to the noise prediction network via:

$$\begin{aligned} \nabla_{z_t} \log p(z_t|y) &\approx -\sigma_t^{-1} \epsilon_\theta(z_t, t, y) \\ \nabla_{z_t} \log p(z_t) &\approx -\sigma_t^{-1} \epsilon_\theta(z_t, t, \emptyset) \end{aligned} \tag{23}$$

where σ_t is the noise magnitude at timestep t. Substituting these identities:

$$\begin{aligned} \nabla_{z_t} \log p(y|z_t) &\approx -\sigma_t^{-1} \epsilon_\theta(z_t, t, y) + \sigma_t^{-1} \epsilon_\theta(z_t, t, \emptyset) \\ &\approx \sigma_t^{-1} \left(\epsilon_\theta(z_t, t, \emptyset) - \epsilon_\theta(z_t, t, y) \right) \\ &\approx \omega \left(\epsilon_\theta(z_t, t, \emptyset) - \epsilon_\theta(z_t, t, y) \right) \end{aligned} \tag{24}$$

where the guidance scale ω absorbs the proportionality constant σ_t^{-1} and sign convention. The final equivalence follows from reordering terms and the scalar nature of ω.

Discussion **Lemma 1** demonstrates that diffusion models can effectively estimate conditional likelihood, thereby providing a foundation for semantic alignment without the need for additional classifier training. In previous works [8, 10, 22, 61], this aspect was incorporated, but without further in-depth analysis. We are the first to elaborate on the design in this aspect and verify its significant impact on the performance of dataset distillation, as shown in Table 3 in the paper.

A2.3: Proof of Proposition 1

Proposition 1 *Given a single step sampling process (such as DDIM) based on ϵ_θ to update $z_{t-1}^{(0)}$ using condition y, consider a dynamic label $\hat{y}_t = y + \delta_t$ where δ_t is a time-dependent vector. The modified sampling step admits the first-order approximation:*

$$z_{t-1} \approx z_{t-1}^{(0)} + \Lambda_t(\delta_t) \tag{25}$$

where the condition shift operator Λ_t is defined as: $\Lambda_t(\delta_t) = c_t \cdot \left(\nabla_y \epsilon_\theta(z_t, t, y)\right)^\top \delta_t$ with $c_t = \sqrt{1-\alpha_{t-1}} - \sqrt{\alpha_{t-1}} \cdot \sqrt{1-\alpha_t}/\sqrt{\alpha_t}$ as the intrinsic time-scaling factor.

Proof. By Taylor expansion, we can approximate the denoising model ϵ under dynamic label.

$$\epsilon_\theta(z_t, t, y + \delta_t) \approx \epsilon_\theta(z_t, t, y) + \nabla_y \epsilon_\theta(z_t, t, y)^\top \delta_t \tag{26}$$

Neglecting the effects of higher-order terms, we substitute the approximation formula into the sampling formula of the diffusion model, taking DDIM (Equation (3)) as an example here:

$$\begin{aligned} z_{t-1} \approx & \alpha_t^1 \left(\frac{z_t - \sqrt{1-\bar{\alpha}_t}(\epsilon_\theta(z_t, t, y) + \nabla_y \epsilon_\theta(z_t, t, y)^\top \delta_t)}{\sqrt{\bar{\alpha}}}\right) \\ & + \alpha_t^2(\epsilon_\theta(z_t, t, y) + \nabla_y \epsilon_\theta(z_t, t, y)^\top \delta_t) + \alpha_t^3 \epsilon \end{aligned} \tag{27}$$

After rearrangement, we obtain:

$$z_{t-1} = z_{t-1}^{(0)} + c_t \nabla_y \epsilon_\theta(z_t, t, y)^\top \delta_t \tag{28}$$

where, $z_{t-1}^{(0)}$ corresponds to a standard DDIM sampling process, $c_t = \sqrt{1-\alpha_{t-1}} - \sqrt{\alpha_{t-1}} \cdot \sqrt{1-\alpha_t}/\sqrt{\alpha_t}$. We define the condition shift operator $\Lambda_t = c_t \nabla_y \epsilon_\theta(z_t, t, y)^\top \delta_t$, which represents the additional shift term introduced by dynamic labels in the sampling dynamics of the data distribution space.

Discussion From Proposition 1, we can observe that the dynamic term introduces an additional shift term into the sampling dynamics of diffusion models. Researchers have demonstrated that such an offset term helps diffusion models move away from local mode points, further explore the distribution space, and thereby enhance diversity [50]. Similarly, adding an shift term directly in the sampling process of the sample space can also achieve a similar effect. However, it should be noted that this method leads to more complex computations due to the higher dimensionality of the sample space. Meanwhile, the regulation of the shift term in the sample space is also trick, unreasonable coefficients may directly disrupt the entire sampling process. Stable regulation coefficients often need to be obtained by computing the derivative of ϵ. Therefore, introducing a dynamic process in the label space is a more reasonable choice for us.

Furthermore, starting from our goal of generating diverse and high-information samples, we propose the design of two shift terms. The noise shift term provides an effective exploration direction, while the soft label term offers guidance toward class boundaries. Since the soft labels selected for each sample are different, the soft label term can also provide reasonable diversity guidance.

A2.4: Proof of Corollary 1

Corollary 1 *Under the conditions of Theorem 1, consider an approximate distribution $\widetilde{P}_\mathcal{T}$ satisfying $W(\widetilde{P}_\mathcal{T}, P_\mathcal{T}) \leq \epsilon$ for small $\epsilon > 0$. Assuming the distance metric satisfies the triangle inequality, distributions lie in a Polish space. The risk discrepancy is bounded by:*

$$|R_\mathcal{T}(\theta_\mathcal{T}^*) - R_\mathcal{T}(\theta_\mathcal{S}^*)| \leq 2L \cdot \Big(W(P_\mathcal{S}, \widetilde{P}_\mathcal{T}) + W(P_\mathcal{T}, \widetilde{P}_\mathcal{T})\Big) \tag{29}$$

Proof. Let $P_\mathcal{S}, P_\mathcal{T}, \widetilde{P}_\mathcal{T}$ be Borel probability measures on a Polish metric space (X, d). Let γ_1 and γ_2 be the optimal couplings corresponding to $W(P_\mathcal{S}, \widetilde{P}_\mathcal{T})$ and $W(P_\mathcal{T}, \widetilde{P}_\mathcal{T})$, respectively. By the **gluing lemma**, construct a measure γ on X^3 with (x, y)-marginal γ_1 and (y, z)-marginal γ_2. Project γ to a coupling $\gamma_{13} \in \Gamma(P_S, P_T)$ via $\gamma_{13}(A \times C) = \gamma(A \times X \times C)$. Then, using the triangle inequality for d, we have:

$$\begin{aligned} \int_{X^2} d(x, z)\, d\gamma_{13} &= \int_{X^3} d(x, z)\, d\gamma \\ &\leq \int_{X^3} [d(x, y) + d(y, z)]\, d\gamma \\ &\leq \int_{X^2} d(x, y)\, d\gamma_1 + \int_{X^2} d(y, z)\, d\gamma_2 \\ &= W(P_\mathcal{S}, \widetilde{P}_\mathcal{T}) + W(P_\mathcal{T}, \widetilde{P}_\mathcal{T}) \end{aligned} \tag{30}$$

Since $W(P_\mathcal{S}, P_\mathcal{T})$ is the infimum over all couplings in $\Gamma(P_S, P_T)$:

$$\begin{aligned} W(P_\mathcal{S}, P_\mathcal{T}) &\leq \int_{X^2} d(x, z)\, d\gamma_{13} \\ &\leq W(P_\mathcal{S}, \widetilde{P}_\mathcal{T}) + W(P_\mathcal{T}, \widetilde{P}_\mathcal{T}) \end{aligned} \tag{31}$$

Substituting the results into the Theorem 1, we can obtain:

$$|R_\mathcal{T}(\theta_\mathcal{T}^*) - R_\mathcal{T}(\theta_\mathcal{S}^*)| \leq 2L \cdot \Big(W(P_\mathcal{S}, \widetilde{P}_\mathcal{T}) + W(P_\mathcal{T}, \widetilde{P}_\mathcal{T})\Big) \tag{32}$$

Disscusion. **Corollary 1** reveals that the target risk discrepancy admits an upper bound. This decomposition provides critical guidance for practical implementation. The first term $W(P_\mathcal{S}, \widetilde{P}_\mathcal{T})$ represents the alignment error, whose optimization requires $\widetilde{P}_\mathcal{T}$ to be computationally tractable.

The second term $W(\widetilde{P}_{\mathcal{T}}, P_{\mathcal{T}})$ quantifies the approximation error, which must be minimized to preserve distributional fidelity. To satisfy both requirements, we seek a discrete approximation $\widetilde{P}_{\mathcal{T}}$ that minimizes $W(\widetilde{P}_{\mathcal{T}}, P_{\mathcal{T}})$ while enabling efficient optimization of $P_{\mathcal{S}}$. This leads naturally to the classical optimal quantization problem [4]. Clustering algorithms are efficient solutions with good convergence properties for this type of problem.

A2.5: Proof of Proposition 2

Proposition 2 *Let $\tilde{P}_{\mathcal{T}}^{(1)}$ denote the mean-matching approximation of $P_{\mathcal{T}}$ defined by a Dirac measure δ_μ concentrated at the mean μ of $P_{\mathcal{T}}$, and $\tilde{P}_{\mathcal{T}}^{(2)}$ denote the proposed approximation constructed via our method with cluster count K. The Wasserstein distance satisfies:*

$$W(P_{\mathcal{T}}, \tilde{P}_{\mathcal{T}}^{(2)}) \leq W(P_{\mathcal{T}}, \tilde{P}_{\mathcal{T}}^{(1)}) \tag{33}$$

Proof. For $\tilde{P}_{\mathcal{T}}^{(1)}$, the optimal transport cost is the integral of distances to μ:

$$W(P_{\mathcal{T}}, \tilde{P}_{\mathcal{T}}^{(1)}) = \int d(x, \mu)\, dP_{\mathcal{T}}(x) \tag{34}$$

For $\tilde{P}_{\mathcal{T}}^{(2)}$, consider transporting mass in cluster C_i to its centroid k_i . The cost of this local plan is:

$$\text{Cost} = \sum_{i=1}^{K} \int_{C_i} d(x, k_i)\, dP_{\mathcal{T}}(x) \tag{35}$$

By the key property of K-means, k_i is the optimal center for C_i , meaning it minimizes the local transport cost:

$$\int_{C_k} d(x, k_i)\, dP_{\mathcal{T}}(x) \leq \int_{C_k} d(x, z)\, dP_{\mathcal{T}}(x), \quad \forall z \in \mathbb{R}^d \tag{36}$$

Setting $z = \mu$ (the mean of $P_{\mathcal{T}}$), we immediately get:

$$\int_{C_i} d(x, k_i)\, dP_{\mathcal{T}}(x) \leq \int_{C_i} d(x, \mu)\, dP_{\mathcal{T}}(x) \tag{37}$$

Summing the inequality over all clusters $i = 1, \ldots, K$, we have:

$$\text{Cost} \leq \sum_{i=1}^{K} \int_{C_i} d(x, \mu), dP_{\mathcal{T}}(x) = \int d(x, \mu)\, dP_{\mathcal{T}}(x) \tag{38}$$

For all transport plans, the optimal transport plan achieves the minimal cost, and thus:

$$W(P_{\mathcal{T}}, \tilde{P}_{\mathcal{T}}^{(2)}) \leq \text{Cost} \leq \int d(x, \mu)\, dP_{\mathcal{T}}(x) = W(P_{\mathcal{T}}, \tilde{P}_{\mathcal{T}}^{(1)}) \tag{39}$$

Discussion. In a more general case, we can analyze the bounds of $W(P_{\mathcal{T}}, \tilde{P}_{\mathcal{T}}^{(2)})$. When effectively evaluating the intrinsic manifold dimension of the data, for a specific K, we can obtain the Wasserstein distance convergence bounds between the approximate distribution based on the K-means algorithm and the original distributions. We recommend referring to **Theorem 5.2** in [4] for more details.

A3: More Implementation Details

A3.1: More Details of Method

We provide detailed specifics of the algorithm rapid reproduction, and we will also open-source the implementation code of the paper after organizing it. **Algorithm 1** formalizes the overall framework of our approach, featuring parallel application of dual-matching guidance at targeted diffusion phases.

Semantic Matching. Building on insights from Yu et al. [76], we partition the diffusion process into three distinct phases for semantic matching: 1) Chaotic Stage: Leveraging pure noise vectors as label proxies to facilitate exhaustive stochastic exploration. 2) Semantic Stage: Employing our proposed dynamic soft labels for guided generation. 3) Refinement Stage: Conducting deterministic sampling with target vectors to ensure semantic alignment.The Fig. 1 intuitively illustrates our dynamic sampling process. The label strategy across stages is mathematically formalized as:

$$\widetilde{y}_t = \begin{cases} n & t \geq t_1 \\ \sqrt{\sigma_t} y + (1 - \sqrt{\sigma_t})(\beta_s y^\star + \beta_n n) & t_2 < t < t_1 \\ y & t \leq t_2 \end{cases} \tag{40}$$

where $n \sim \mathcal{N}(0, I)$ denotes Gaussian noise, $y^\star$ is a random select label subject to $y^\star \neq y$, β_n and β_s are modulation coefficients, and σ_t represents a time-dependent scheduling term defined as:

$$\sigma_t = \frac{t_1 - t}{t_1 - t_2} \tag{41}$$

Based on the observations of the stages of the diffusion model, we set $t_1 = 45$ and $t_2 = 25$. Furthermore, to maintain semantic consistency, we also perform rescaling on the label vectors. The rescaling process is determined by the mean and variance of the target label vectors.

$$\widetilde{y}_{re} = \frac{\widetilde{y} - mean(\widetilde{y})}{std(\widetilde{y})} * std(y) + mean(y) \tag{42}$$

Substituting the label vector into the denoising model and applying classifier-free guidance, we have

$$\begin{aligned} \hat{\epsilon}_\theta(z_t, t, \widetilde{y}_t) &= \epsilon_\theta(\mathbf{z}_t, t, \widetilde{y}_t) - \nabla_{\boldsymbol{z}_t} \log p(y|\boldsymbol{z}_t) \\ &\approx (1 + \omega)\epsilon_\theta(\mathbf{z}_t, t, \widetilde{y}_t) - \omega\epsilon_\theta(\mathbf{z}_t, t, \emptyset) \end{aligned} \tag{43}$$

Algorithm 1 Dual Matching-Guided Diffusion Models

Require: CFG factor ω, semantic matching coefficient β_n and β_s, distribution matching coefficient ρ, number of Support Points K, number of class C, image per class N, distribution matching range $[t_1, t_2]$
Require: Target dataset $\mathcal{T}$, pre-trained diffusion model ϵ_θ, VAE decoder model V_D.
Ensure: Surrogate dataset $\mathcal{S}$

for $c = 1$ **to** C **do**
  Obtain the approximated distribution $\widetilde{P}_{\mathcal{T}}$ via Algorithm 2
  Initialize class-aware surrogate dataset storage $\mathcal{S}^c_{[0]} \leftarrow \{\}$
  for $n = 1$ **to** N **do**
    Sample initial random noise $z_n^T \sim N(0, I)$;
    Select $y^\star$, s.t. $y^\star \neq c$
    for $t = T$ **to** t **do**
      Obtain dynamic label $\widetilde{y}_t$ via Equation (40)
      Semantic matching guided sampling $z_{t-1} = D(z_t, t, \widetilde{y}_t, \epsilon_\theta)$.
      if $t \in [t_1, t_2]$ **then**
        Obtain a temporary class-aware surrogate dataset $\mathcal{S}^c_{[n]} \leftarrow \mathcal{S}^c_{[n-1]} \cup \{z_{0|t}(z_t)\}$.
        Calculate the OT loss $\mathcal{L}_{\text{OT}}(P_{\mathcal{S}^c_{[i]}}, \widetilde{P}_{\mathcal{T}^c})$ via Sinkhorn algorithm.
        Distribution matching guided sampling $z_{t-1} = z_{t-1} - \rho_t \nabla_{z_t} \mathcal{L}_{\text{OT}}(P_{\mathcal{S}^c_{[n]}}, \widetilde{P}_{\mathcal{T}^c})$.
      end if
    end for
    Store surrogate data $\mathcal{S}^c_{[n]} \leftarrow \mathcal{S}^c_{[n-1]} \cup \{z_0\}$
  end for
end for
return Decoded synthetic image $\mathcal{S} = V_D(\mathcal{S}_{[N]})$

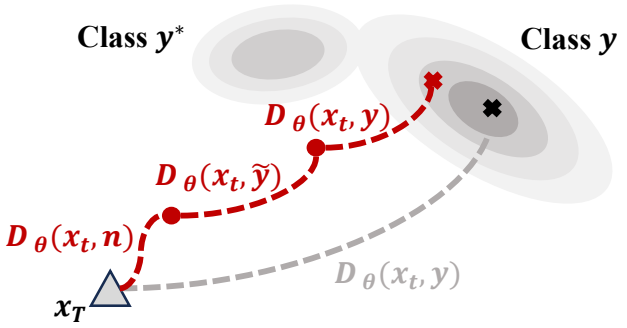


Figure 1. Intuitive demonstration of the dynamic semantic matching guided sampling process. Compared with the sampling process without dynamic guidance, our method can greatly improve diversity and avoid oversampling samples in high-density regions.

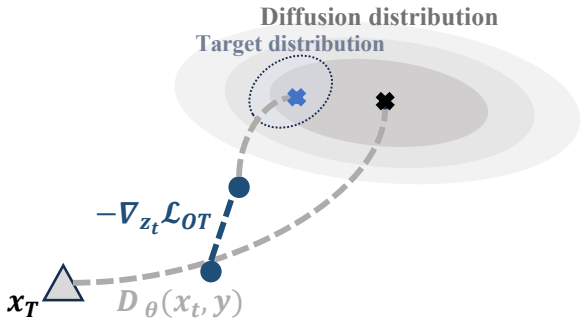


Figure 2. Intuitive demonstration of the distribution matching guided sampling process.Compared with the original sampling process , adding distribution matching guidance can enable the sampling regions to align with the distribution of the target dataset, which is particularly applicable when there is a large discrepancy between the distribution of the diffusion model and that of the target dataset.

In practice, $\emptyset$ is also injected into the denoising model in the form of label vectors. Therefore, we suggest imposing a dynamic process on it as well to improve stability. We define the single-step dynamic sampling process as $z_{t-1} = D_\theta(z_t, t, \widetilde{y}_t)$.

Distribution Matching. We introduce the distribution matching term in parallel to guide sampling, which show in Fig. 2. Distribution matching is performed within each class to avoid introducing additional semantic information that interferes with semantic alignment. First, we map the samples in the target dataset to the latent space of the diffusion model. Given the VAE encoder V_E and the image sample $x^i \sim P_{\mathcal{T}}$ and $x^i \in \mathbb{R}^D$, we have:

$$z_0^i = V_E(x^i) \tag{44}$$

Where, $z \in \mathbb{R}^N$ with $N < D$. To distinguish from noise samples, we define the samples from the data distribution in the latent space as z_0. We also introduce a hyperspherical projection to project latent space samples onto the hypersphere of $\mathbb{R}^{N-1}$:

$$\hat{z}_0^i = \frac{z_0^i}{||z_0^i||_2} \tag{45}$$

We define the Euclidean distance in the latent space as the distance metric, which is used for distribution approximation and subsequent optimal transport.

$$d(z_0^i, z_0^j) = ||z_0^i - z_0^j||_2 \tag{46}$$

Before performing distribution matching, we first need to perform distribution approximation on the distribution of the target dataset to optimize the efficiency of computing optimal transport. We adopt an implementation of the GPU-based fast K-means clustering algorithm [46]. Taking the centroid of the cluster as support points of the approximate distribution, the normalization of the cardinality of each cluster serves as the mass coefficients. We present our distribution approximation algorithm in **Algorithm 2**.

Algorithm 2 K-Means based Distribution Approximation

Require: Target dataset $\mathcal{T}$, cluster count K
Ensure: Approximated distribution $\widetilde{P}_{\mathcal{T}}$

```
Initialize centroids {k_i^(0)}_{i=1}^K
for iter = 1 to max_iter do
    C_i^(iter) ← {x ∈ T : arg min_j ‖x − k_j^(iter−1)‖}
    k_i^(iter) ← 1/|C_i^(iter)| Σ_{x∈C_i^(iter)} x
end for
Compute masses: m_i ← |C_i^(final)|/|T|
return P̃_T = Σ_{i=1}^K m_i δ_{k_i}
```

Since optimizing all instances simultaneously in the diffusion model space is not feasible, we propose a greedy progressive matching strategy. We construct a memory set $\mathcal{S}_{[n]}^c = \{z_0^i, y^i\}_{i=1}^n$ to store the generated surrogate samples,where $y^i = c$. For the next surrogate sample, we first initialize it from the noise distribution $z_T^{n+1} \sim N(0, I)$ and execute the reverse process. Notably, applying distribution matching guidance in all sample stages is unnecessary (see Table 7 in the Appendix A4), so we only perform it in stages where $t \in [30, 45]$. For a z_t^{n+1} to be optimized, we first map it to the clean data distribution through single-step diffusion.

$$z_{0|t}^{n+1} = \frac{z_t^{n+1} - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(z_t^{n+1}, t, c)}{\sqrt{\bar{\alpha}_t}} \tag{47}$$

We construct a temporary distribution $\mathcal{S}_{[n+1]}^c$ by combining $z_{0|t}^{n+1}$ with $\mathcal{S}_{[n]}^c$, and calculate the optimal transport distance in sample distribution $P_{\mathcal{S}_{[n+1]}^c}$.

$$\mathcal{L}_{\text{OT}} = W(P_{\mathcal{S}_{[n+1]}^c}, P_{\mathcal{T}^c}) = \langle \gamma, \mathbf{C} \rangle \tag{48}$$

where γ is the optimal coupling, C is the cost matrix. We only need to focus on z^{n+1}, which means that $\mathcal{L}_{\text{OT}}$ can be simplified to:

$$\mathcal{L}_{\text{OT}} = \sum_{j=1}^{K} \gamma^{n+1,j} \cdot C^{n+1,j} \tag{49}$$

This loss models the matching relationship between the new sample z^{n+1} and the support points of the approximate distribution. Due to the presence of optimal transport, the optimization direction of $\mathcal{L}_{\text{OT}}$ will prompt z^{n+1} to align with the unaligned regions in the approximate distribution, ensuring the performance of distribution alignment. We utilize training-free guidance technology [74, 76] to incorporate this loss into the diffusion model framework.

$$\begin{aligned} z_{t-1}^{n+1} &= D_\theta(z_t^{n+1}) - \rho_t \nabla_{z_t^{n+1}} \mathcal{L}_{\text{OT}}(P_{\mathcal{S}_{[n+1]}^c}, P_{\mathcal{T}}) \\ &= D_\theta(z_t^{n+1}) - \rho_t \nabla_{z_t^{n+1}} \sum_{j}^{K} \gamma^{n+1,j} \cdot C^{n+1,j} \end{aligned} \tag{50}$$

Following the suggestions from previous work [74], we set ρ_t as a time-dependent term and a scale term ρ:

$$\rho_t = \rho * log\alpha_t^2 \tag{51}$$

$log\alpha_t^2$ is log-variance of diffusion models.

Diversity vs. Representativeness trade-off While representativeness and diversity have been demonstrated as two crucial characteristics for dataset distillation optimization [61], our semantic matching and distribution matching specifically enhance representativeness in the semantic space and sample space respectively. Simultaneously, our dynamic guidance mechanism provides a pathway for further diversity optimization. However, in practice, we observe that diversity enhancement does not consistently translate into performance gains. As shown in Table 7 and Table 8, we find that for lower IPC settings, diversity enhancement may be unnecessary, whereas in higher IPC configurations, it facilitates broader exploration of the generative distribution and helps avoid local optima. This explains the more substantial performance improvements observed under high IPC conditions. We plan to formulate diversity-related parameters as functions of IPC, representing a promising direction for future research.

A3.2: More Details of Experiment setting

Evaluation Protocol In the hard label protocol, we follow Gu et al. [22] original code and parameter definitions;

for more details, please refer to their work. During the training of the target network, we apply the same random resize-crop and CutMix as data augmentation techniques. This protocol is used to evaluate ImageNet subsets, including ImageNet-Woof and ImageNet-Nette [27]. It should be noted that we also applied the hard label protocol to ImageNet-IDC [28]; however, to align with Gu et al. [22] and Kim et al. [28], our specific settings adopt Kim et al. [28]'s definitions.

In the soft label protocol, we followed Sun et al. [61] original code and parameter settings. Soft labels are generated by a pre-trained ResNet-18 [24] model and applied to the training of the evaluation model. We applied this protocol to the full ImageNet-1k dataset.

Under the same experimental settings and repeated experiments, we directly report the experimental results of previous works [8, 22, 28, 61].

Evaluation metric We have provided explanations for the evaluation metrics adopted in the paper, including:

- **Accuracy:** The accuracy metric denotes the best TOP-1 accuracy on the test set achieved during the training process of the evaluation model. For the evaluation model, we repeated the training 5 times and report the mean and standard deviation of the best TOP-1 accuracy.
- **Coverage:** For the coverage metric, we first extract features of the dataset using a pre-trained VGG network [58]. The feature space we adopt is the output of the second classification layer of the VGG network, which has a total of 4096 dimensions. We performed code in Naeem et al. [45] to calculate the coverage between the surrogate dataset and the target dataset.
- **Optimal Transport Dataset Distance:** We utilized the idea of Alvarez-Melis and Fusi [2] and calculated the optimal transport between datasets to evaluate the dataset distance. We adopted the same VGGnet feature space and used t-SNE [42] to map it to a two-dimensional space. Optimal transport was applied in the t-SNE space to calculate the dataset distance [19].
- **Diversity:** For the diversity metric, we calculate the minimum distance between each surrogate sample and all other surrogate samples in the VGG feature space, and report the average as the diversity metric.
- **FID:** We directly adopted the official implementation of clean-fid to calculate the FID scores between the surrogate dataset and the target dataset [47].
- **Other generative quality metrics**: We directly applied the official implementations of [45] in the VGG feature space.

Method	IPC-10	IPC-20	IPC-50
Random	$48.1_{\pm 0.8}$	$52.5_{\pm 0.9}$	68.1 ± 0.7
DM [82]	$52.8_{\pm 0.5}$	$58.5_{\pm 0.4}$	$69.1_{\pm 0.8}$
DiT [48]	$54.1_{\pm 0.4}$	$58.9_{\pm 0.2}$	$64.3_{\pm 0.6}$
D^4M [60]	$51.1_{\pm 2.4}$	$58.0_{\pm 1.4}$	$64.1_{\pm 2.5}$
Minimax [22]	$53.1_{\pm 0.2}$	$59.0_{\pm 0.4}$	$69.6_{\pm 0.2}$
MGD^3 [8]	$55.9_{\pm 0.4}$	$61.9_{\pm 0.9}$	$72.1_{\pm 0.8}$
Ours	$\mathbf{57.0}_{\pm 0.3}$	$\mathbf{63.3}_{\pm 1.4}$	$\mathbf{73.2}_{\pm 0.7}$

Table 1. Performance comparison between our method and state-of-the-art methods on ImageNet-IDC10, evaluated under the hard-label protocol of Kim et al. [28]. Results are reported as Top-1 accuracy on ResNet-AP with average pooling. The best performance is highlighted in **bold**.

A4: Additional Result

A4.1: Experiments on Imagenet-IDC

We conducted additional experiments on ImageNet-IDC. ImageNet-IDC is a dataset consisting of 100 classes, among which Imagenet-IDC10 corresponds to the first ten classes [28]. We adopted the hyperparameters defined on ImageNet-Woof without additional parameter tuning. The Table 1 presents our results on ImageNet-IDC10. Our method achieved the best performance across all IPC settings. Compared with the SOTA method MGD3, we achieved improvements of 1.1%, 1.4%, and 1.1% respectively. This further validates the generalization capability of our proposed framework.

ImageNet-IDC100 is a more complex and larger-scale subset. The Table 2 presents our comparative experiments on it. Our method achieves performance comparable to state-of-the-art (SOTA) models while being more stable. This also validates the scalability of our proposed framework. Further precise parameter tuning could yield better results, but we have not conducted it due to constraints on computational resources.

Overall, our method achieves excellent performance on the ImageNet-IDC dataset. Compared with other dataset distillation algorithms, our method is more efficient. Under the IPC-10 setting for Imagenet-IDC100, IDC-1 [28] requires over 100 hours of optimization time, while Minimax [22] needs nearly 10 hours for fine-tuning. In contrast, our method introduces only a small amount of additional computation during the sampling stage and takes approximately 0.5 hours.

A4.2: Experiments on Imagenet-A,B,C,D,E

We conducted further comparisons using the evaluation benchmark provided by LD3M[44]. This benchmark comprises five ImageNet subsets, designated as A, B, C, D, and E. Evaluations were performed across four distinct network

Method	IPC-10	IPC-20
Random	$19.1_{\pm 0.4}$	$26.7_{\pm 0.5}$
Herding [69]	$19.8_{\pm 0.3}$	$27.6_{\pm 0.1}$
IDC-1 [28]	$\underline{25.7}_{\pm 0.1}$	$29.9_{\pm 0.2}$
Minimax [22]	$24.8_{\pm 0.2}$	$32.3_{\pm 0.1}$
MGD^3 [8]	$\mathbf{25.8}_{\pm 0.5}$	$\underline{33.9}_{\pm 1.1}$
Ours	$\underline{25.7}_{\pm 0.4}$	$\mathbf{34.0}_{\pm 0.1}$

Table 2. Performance comparison between our method and state-of-the-art methods on ImageNet-IDC100 , evaluated under the hard-label protocol of Kim et al. [28]. Results are reported as Top-1 accuracy on ResNet-AP with average pooling. The best performance is highlighted in **bold**, while the second-best is underlined.

architectures (AlexNet, VGG11, ResNet18, and ViT), reporting the mean top-1 accuracy over 5 runs. We maintained the same hyperparameter configuration as ImageNet-Woof. The IPC-10 evaluation results are presented in the Table 3, where our model achieves comprehensive superiority across all subsets. These results collectively demonstrate the strong cross-dataset and cross-architecture generalization capability of our method.

A4.3: Additional Comparisons on ConvNet-6

We further evaluate our method using ConvNet-6 under various IPC settings. Our approach achieves superior performance at IPC-20 and IPC-50 configurations. However, at IPC-10, our results fall slightly behind Minimax, which can be attributed to our use of the same hyperparameter configuration as employed for IPC-50. Subsequent experiments revealed that allocating distinct hyperparameters for different IPC settings yields more optimal results. This is because lower IPC settings require less emphasis on diversity enhancement while prioritizing the generation of more representative samples. We provide comprehensive analysis and discussion of this aspect in Table 7 and Table 8.

A4.4: Additional Ablation Study

In this subsection, we conduct more detailed ablation study to demonstrate the rationality of the design of our method. We focus on three key aspects of the method design: 1) the construction mechanism of dynamic labels; 2) the impact of different distribution approximation algorithms; 3) the guidance stage of matching terms.

Construction Mechanism of Dynamic Labels We compared the construction method using only random noise (Noise) with the dynamic soft label construction method we adopted (Soft label with Noise). The results are presented in Table 5. We found that constructing dynamic labels using only noise terms can also achieve effective performance gains; particularly under the IPC-50 setting, it achieves performance close to that of our full method. This further demonstrates the excellent performance of our proposed dynamic label semantic matching technique in enhancing the diversity of dataset distillation. Moreover, after adding soft label terms, the performance can be further improved, which illustrates the effectiveness of the deterministic shift term we defined for the dataset distillation task. This experiment also proves that designing effective semantic matching guidance is one of the key factors for enhancing dataset distillation performance, which has often been overlooked in previous work.

We further conducted an analysis by combining distribution matching (OT), and it can be observed that dataset distillation performance is further enhanced after integrating distribution matching. This also experimentally validates our proposed theoretical framework (**Theorem 1**).

Different Distribution Approximation Distribution approximation is a key component of our algorithm, and its performance influences the performance of distribution matching based on optimal transport. We conducted an ablation study on three different distribution approximation algorithms. The widely used classical distribution matching loss [82], mean matching (mean), can be regarded as a special case of our proposed method when $K = 1$. This represents allocating the mass of the entire conditional distribution to the distribution center. Density-based random sampling (DBS) is a sampling-based distribution approximation method, which selects support points by calculating the density of sample points from the original distribution to assign sampling probabilities, and normalizes the densities of different support points as mass coefficients. In the Tab. 6, we present the performance comparison of the three methods.

The Mean achieves effective performance gains under IPC-10. However, in high IPC settings, the Mean yields overly coarse distribution approximations and fails to fully model the distribution structure. Meanwhile, for diffusion models that can only optimize a single sample at a time, matching against the same mean point impairs diversity. This experimental result also validates our Proposition 2.

DBS provides effective signals for distribution matching at high IPC settings. Nevertheless, due to its randomness, DBS often fails to capture comprehensive representative points and particularly overlooks some fine-grained patterns with small K. This randomness impairs dataset distillation performance, especially under low IPC settings.

The comprehensively leading performance results of our proposed method demonstrate the effectiveness of our proposed local distribution approximation matching. Notably, our distribution approximation technique provides an efficient solution for achieving distribution alignment with limited samples under resource-constrained settings. Com-

Distil Alg.	Method	ImageNet-A	ImageNet-B	ImageNet-C	ImageNet-D	ImageNet-E
DC	Pixel	52.3±0.7	45.1±8.3	40.1±7.6	36.1±0.4	38.1±0.4
	GLaD[7]	53.1±1.4	50.1±0.6	48.9±1.1	38.9±1.0	38.4±0.7
	LD3M[44]	55.2±1.0	51.8±1.4	49.9±1.3	39.5±1.0	39.0±1.3
DM	Pixel	52.6±0.4	50.6±0.5	47.5±0.7	35.4±0.4	36.0±0.57
	GLaD[7]	52.8±1.0	51.3±0.6	49.7±0.4	36.4±0.4	38.6±0.7
	LD3M[44]	57.0±1.3	52.3±1.1	48.2±4.9	39.5±1.5	39.4±1.8
	MGD^3[8]	63.4±0.8	66.3±1.1	58.6±1.2	**46.8**±0.8	51.1±1.0
	Ours	**65.4**±0.3	**70.2**±0.9	**62.2**±1.0	**46.8**±1.5	**51.3**±0.3

Table 3. Performance comparison between our method and state-of-the-art methods on ImageNet-A, B, C, D, and E , evaluated via the benchmark of Moser et al. [44] under IPC-10. Results are reported as mean Top-1 accuracy on AlexNet, VGG11, ResNet18, and ViT. The best performance is highlighted in **bold**.

Method	IPC-10	IPC-20	IPC-50
Random	24.3±1.1	29.1±0.7	41.3±0.6
DM [82]	26.9±1.2	29.9±1.0	44.4±1.0
Glad [7]	33.8±0.9		
IDC-1[28]	33.3±1.1	35.5±0.8	43.9±1.2
DiT [48]	34.2±0.4	36.1±0.8	46.5±0.8
Minimax [22]	**37.0**±1.0	37.6±0.2	53.9±0.6
MGD^3 [8]	34.7±1.1	39.0±3.5	54.5±1.6
Ours	34.5±0.1	**40.1**±0.3	**54.9**±0.5

Table 4. Performance comparison between our method and state-of-the-art methods on ImageNet-Woof. Results are reported as Top-1 accuracy on ConvNet-6. The best performance is highlighted in **bold**.

Dynamic label	IPC-10	IPC-50
DiT	34.7±0.5	49.3±0.2
Noise	36.6±1.7	59.2±1.1
Noise + OT	40.6±1.4	59.7±0.3
Soft label with Noise	38.9±1.2	59.3±0.4
Soft label with Noise + OT	**40.8**±1.2	**60.1**±0.8

Table 5. Ablation study on different dynamic label construction methods. Results are reported as Top-1 accuracy on ResNet-10 with average pooling in ImageNet-Woof. The best performance is highlighted in **bold**.

Approximation	IPC-10	IPC-50	IPC-100
Mean	<u>39.6</u>±1.1	58.6±0.5	62.5 ±0.4
DBS	39.2±1.6	<u>59.6</u>±0.5	<u>64.4</u> ±0.5
K-means	**40.8**±1.2	**60.1**±0.8	**65.8** ±0.2

Table 6. Ablation study on different distribution approximation methods. Results are reported as Top-1 accuracy on ResNet-10 with average pooling in Imagenet-Woof. The best performance is highlighted in **bold**, while the second-best is <u>underlined</u>.

Guidance Mechanism	IPC-10	IPC-50
Semantic matching		
Dynamic Soft Label	35.1±0.8	55.6±0.4
w/ Stochastic Exploration	31.2±0.8	54.3±1.5
w/ Semantic Refinement	**42.0**±1.5	<u>59.6</u>±1.6
Distribution matching		
Full-stage guidance	40.4±0.5	57.2±0.7
Ours Full	<u>40.8</u>±1.1	**60.1**±0.8

Table 7. Ablation study on different guidance mechanism. Results are reported as Top-1 accuracy on ResNet-10 with average pooling in Imagenet-Woof. The best performance is highlighted in **bold**, while the second-best is <u>underlined</u>.

pared to MGD^3 which requires sample sizes proportional to IPC, our method maintains stable performance with fixed-size samples while achieving effective performance gains even at high IPC settings (e.g., IPC=100).

Guidance Mechanism We explored the guidance mechanism, and the results are presented in the Table 7. For semantic matching, we found that the Semantic Refinement is necessary. Using only dynamic soft labels or only combining with Stochastic Exploration leads to performance degradation due to insufficient semantic alignment. This further illustrates the criticality of our proposed semantic alignment assumption. However, incorporating the Semantic Refinement can further ensure semantic alignment and improve dataset distillation performance. Especially under lower IPC settings, Dynamic Soft Label combining only Semantic

Hyperparameter	IPC-10	IPC-50
$\beta_n = 0.01$	$42.7_{\pm 1.3}$	$58.7_{\pm 0.7}$
$\beta_n = 0.04$	$40.5_{\pm 0.2}$	$60.1_{\pm 0.7}$
$\beta_n = 0.1$	$36.3_{\pm 0.6}$	$60.2_{\pm 0.7}$
$\beta_s = 0.04$	$41.5_{\pm 0.5}$	$59.9_{\pm 0.7}$
$\beta_s = 0.06$	$41.2_{\pm 0.5}$	$59.7_{\pm 1.3}$
$\beta_s = 0.1$	$41.3_{\pm 1.1}$	$59.9_{\pm 0.8}$
$1 + \omega = 1$	$19.9_{\pm 0.5}$	$38.6_{\pm 1.2}$
$1 + \omega = 7$	$38.9_{\pm 1.1}$	$57.6_{\pm 1.6}$
$tw = [20, 45]$	$40.6_{\pm 0.8}$	$60.4_{\pm 0.9}$
$tw = [30, 45]$	$40.4_{\pm 1.0}$	$59.6_{\pm 0.6}$
$tw = [25, 40]$	$39.8_{\pm 0.4}$	$59.9_{\pm 0.6}$
$tw = [25, 50]$	$42.0_{\pm 1.5}$	$59.6_{\pm 1.6}$
Ours	$40.8_{\pm 1.1}$	$60.1_{\pm 0.8}$

Table 8. Evaluation of different parameter. Results are reported as Top-1 accuracy on ResNet-10 with average pooling in ImageNet-Woof.

Refinement achieves optimal performance. In higher IPC settings, further introducing Stochastic Exploration can enhance performance, from 59.6% to 60.1%. This empirically validates our earlier discussion: stochastic exploration for diversity enhancement becomes superfluous under low IPC settings, since dynamic labeling already provides sufficient diversity improvement. Therefore, we can further optimize performance by adaptively controlling the temporal window size for stochastic exploration in accordance with the IPC configuration.

For distribution matching, we investigated the differences between full-stage guidance and the key-stage guidance we adopted. We found that full-stage guidance fails to improve performance; on the contrary, in high IPC stages, it may significantly impair performance. This is because in the early stage of sampling, samples have not generated sufficient semantic information, and guidance through distribution matching at this point will produce erroneous guidance signals. Meanwhile, in the later stage, gradient-based guidance may also introduce artifacts into the images, so guidance should be terminated early. Our observations on loss values also illustrate this point: $\mathcal{L}_{\text{OT}}$ decreases only in the key-stage. Therefore, performing distribution guidance only in the key-stage is a reasonable and efficient choice.

A4.5: Additional Hyperparameter Analysis

We analyzed the hyperparameters involved in semantic matching, including CFG scale $1 + \omega$, temporal window tw, modulation coefficient β_n and β_s. Results of the hyperparameter analysis are presented in the Table 8.

β_n is used to regulate the intensity of the noise term. A larger β_n will result in stronger stochastic exploration and further enhance the diversity of generation. Therefore, a larger β_n can enhance performance under high IPC settings but may also impair performance under low IPC settings. Specifically, we found that under low IPC settings, $\beta_n = 0.01$ achieves optimal performance. This validates our assumption that under low IPC settings, randomness should be reduced to generate representative key samples, whereas under high IPC settings, greater consideration of diversity is needed to achieve better performance.

Our analytical experiments on β_s further demonstrate the role of our soft label terms. We found that under low IPC settings, using stronger guidance via soft label terms can generate more informative samples, thereby further improving performance. Meanwhile, under high IPC settings, soft label terms of different intensities can all ensure stable performance. Under different IPC settings, appropriately adjusting β_s and β_m is undoubtedly a better choice. We recommend that for low IPC settings, β_s can be increased while β_n is decreased to generate representative samples with high information concentration. In contrast, under high IPC settings, we focus more on enhancing diversity, and increasing β_n will further strengthen performance. To demonstrate the generality of our method across different IPC settings, we adopted unified parameters $\beta_s = 0.01$ and $\beta_n = 0.06$, which still achieve SOTA performance.

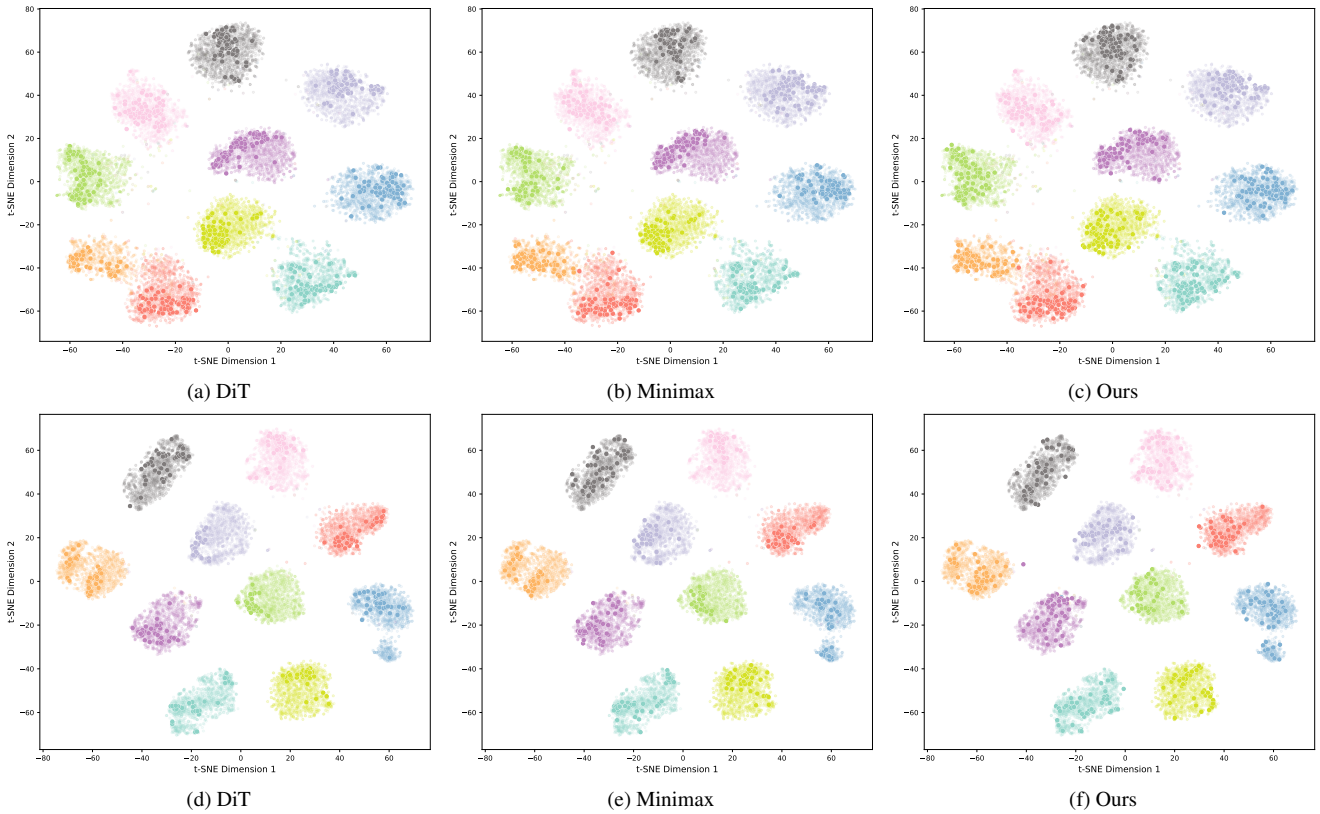
The parameter $1 + \omega$ controls the strength of semantic matching. When $1 + \omega = 1$, the diffusion model fails to capture sufficient semantic information, resulting in significantly degraded performance. Conversely, at $1 + \omega = 7$, overly strong semantic alignment impairs generation quality, also leading to decreased performance. Therefore, we set $1 + \omega = 4$.

Additionally, we performed hyperparameter validation on the temporal window using a step size of 5. We observed that within small variation ranges, the temporal window exhibits minimal impact on the overall experimental results, which aligns with our hypothesis regarding the diversity-representativeness trade-off. For more extreme variations, we have conducted corresponding experiments as documented in Table 7. The selection of this parameter is informed by empirical observations of the diffusion model sampling process in prior literature [76], with its effectiveness further verified through our sensitivity analysis.

A4.6: Generation Quality Evaluation

We evaluated the approach using additional generative quality metrics. Results are presented in the Table 9. On common generative quality metrics, our method achieves performance comparable to the original DiT, demonstrating the realism of our generated samples.

Notably, we observe that diversity enhancement inevitably incurs a slight compromise in semantic representa-



(a) DiT (b) Minimax (c) Ours

(d) DiT (e) Minimax (f) Ours

Figure 3. **Distribution Visualization:** Visualization results of sample distributions for surrogate datasets generated by different methods and the original dataset: top row corresponds to ImageNet-Woof under IPC-100 setting, bottom row corresponds to ImageNet-Nette under IPC-50 setting.

tiveness, reflected by marginally reduced precision. However, the substantially improved recall demonstrates our method's enhanced diversity, which ultimately translates into performance gains. As previously discussed, this minor representational trade-off becomes negligible under high IPC settings. Furthermore, such representational degradation can be effectively mitigated by incorporating soft-label criteria during subsequent distillation stages.

Method	DiT	Minimax	**Ours**
FID	48.6	49.2	48.8
Precision (%)	91.2	94.4	92.4
Recall (%)	51.2	49.5	57.8
Density (%)	1.19	1.38	1.36

Table 9. Evaluation of generation quality evaluation of 10 classes each with 100 images in ImageNet-Woof.

A4.7: Visualization

Generated Samples Visualization: We present generated samples for visualization. Fig. 5 and Fig. 6 present generated examples on the ImageNet-Nette and ImageNet-Woof datasets, respectively. The generated samples were randomly selected under the IPC-50 setting and arranged from left to right in the order of generation. This visualization demonstrates our method's intra-class diversity: earlier samples exhibit stronger semantic representativeness, while later samples display greater uniqueness.

Distribution Visualization: We further visualize the sample distributions using t-SNE. Figure 3 presents the distributions of DiT, Minimax, and our method within the same feature space (Inception-v3) on Imagenet-woof and Imagenet-Nette, demonstrating our approach's diverse semantic matching capability and effective distribution alignment. Our method achieves superior coverage of the target dataset's distribution.

OT Distance Visualization: To better illustrate the relationship between distribution matching and semantic matching, we visualize the optimal transport (OT) distance losses under two scenarios: using distribution matching alone (Distribution Matching), and combining distribution

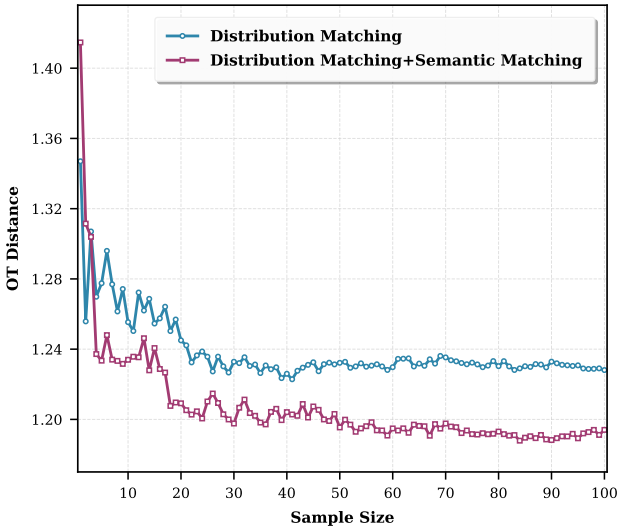


Figure 4. **OT Distance Visualization:** We systematically recorded the final optimal transport (OT) distance loss for each sample during progressive distillation. A randomly selected category from ImageNet-Woof was visualized to illustrate the results.

matching with diversity-enhanced semantic matching (Distribution Matching+Semantic Matching). Each data point represents the final OT distance loss of an individual sample during progressive distillation. The visualization results for a randomly selected class from ImageNet-Woof are presented in the Figure 4. It can be observed that our distribution matching module effectively optimizes the OT loss during dataset distillation. However, due to the diffusion models tendency to generate homogeneous samples, this optimization is prone to converge to local optima. In contrast, the diversity-enhanced semantic matching provides superior distribution exploration capability, which not only accelerates OT loss optimization but also alleviates the local optimum problem. These findings validate that our proposed dual-matching framework exhibits no optimization conflicts, thereby further demonstrating the effectiveness of DMGD.

6.1. Limitation

Currently, our method is confined to dataset distillation with limited semantic scope. Exploration regarding diffusion models possessing general semantic properties and more complex datasets remains insufficient. Furthermore, due to inherent constraints of diffusion models, our approach can not generalize to other data modalities, such as audio [9], video [41], time-series [53] or embodied AI data [38]. In future work, we aim to push the boundaries of dataset distillation towards a more universal and efficient paradigm.

Figure 5. Generated samples are from our proposed DMGD method for the ImageNet-Woof dataset. We present the randomly selected generated samples under the IPC-50 setting. The class names are marked at the left of each row.

Figure 6. Generated samples are from our proposed DMGD method for the ImageNet-Nette dataset. We present the randomly selected generated samples under the IPC-50 setting. The class names are marked at the left of each row.