arXiv:2412.09959v5 [cs.CV] 25 Jun 2026

Towards Consistent and Efficient Dataset Distillation via Diffusion-Driven Selection

Xinhao Zhong[1⋆], Shuoyang Sun[1*], Zhaoyang Xu[1*], XUlin Gu[2], Bin Chen[1⋆⋆], Min Zhang[1], and Yaowei Wang[13]

[1] Harbin Institute of Technology, Shenzhen
[2] Shanghai Jiao Tong University
[3] Pengcheng Laboratory

Abstract. Dataset distillation provides an effective approach to reduce memory and computational costs by optimizing a compact dataset that achieves performance comparable to the full original. However, for large-scale datasets and complex deep networks (e.g., ImageNet-1K with ResNet-101), the vast optimization space hinders distillation effectiveness, limiting practical applications. Recent methods leverage pre-trained diffusion models to directly generate informative images, thereby bypassing pixel-level optimization and achieving promising results. Nonetheless, these approaches often suffer from distribution shifts between the pre-trained diffusion prior and target datasets, as well as the need for multiple distillation steps under varying settings. To overcome these challenges, we propose a novel framework that is orthogonal to existing diffusion-based distillation techniques by utilizing the diffusion prior for patch selection rather than generation. Our method predicts noise from the diffusion model conditioned on input images and optional text prompts (with or without label information), and computes the associated loss for each image-patch pair. Based on the loss differences, we identify distinctive regions within the original images. Furthermore, we apply intra-class clustering and ranking on the selected patches to enforce diversity constraints. This streamlined pipeline enables a one-step distillation process. Extensive experiments demonstrate that our approach consistently outperforms state-of-the-art methods across various metrics and settings. The source code is available at `https://github.com/ndhg1213/DDDS`

Keywords: Dataset distillation · Diffusion model

1 Introduction

The rapid advancement of deep learning has driven impressive performance gains through increasingly deeper and more complex models trained on large-scale datasets [6,10]. However, training these models demands significant memory and computational resources. Dataset distillation has recently emerged as an effective approach to data compression [18], offering reductions in resource consumption.

⋆ Equal Contribution
⋆⋆ Corresponding Author

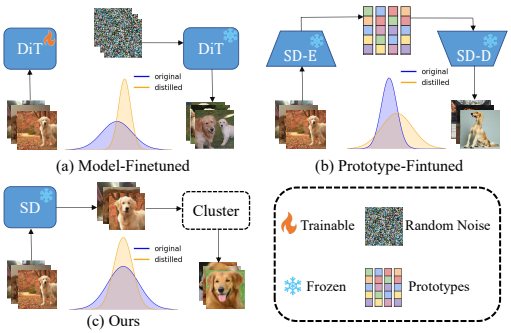


Fig. 1: Comparison of diffusion-based distillation methods. (a) Fine-tuning a diffusion model pre-trained on ImageNet-1K to directly generate images. (b) Fine-tuning the prototypes of the original images to generate images. (c) Identifying the most representative regions in the real images. The images generated by our method more effectively represent the feature distribution of the original dataset.

By optimizing images in pixel space through information matching (e.g., gradient matching), these optimization-based methods produce smaller synthetic datasets that can achieve comparable performance to the original data [1, 9, 34, 35, 37]. Despite their promising performance, the vast pixel-level optimization space restricts the feasibility of these methods to small-scale datasets, e.g., CIFAR-10/100 [16], thus limits their generalization. A recent approach, SRe^{2}L [33], scales optimization-based methods to larger datasets (e.g., ImageNet-1K [5]) by using a pre-trained classifier to guide dataset synthesis. However, its strong dependence on proxy models during optimization often hinders the distilled dataset's generalization, limiting broader applicability. Another critical limitation arises when handling various distillation configurations, such as different subsets of large datasets. Previous optimization-based methods typically incur substantial redundant costs due to repeated distillation requirements. Although recent works [11,12] aim to address this issue, their applicability remains confined to small-scale datasets.

Subsequent methods [3, 4, 8, 29, 30, 36] that avoid direct image optimization have shown promise for scaling dataset distillation to larger datasets and more complex model architectures, addressing several severe limitations of previous methods. RDED [30] leverages a pre-trained classifier (e.g., ResNet-18 [10]) to select patches randomly cropped from real images and then concatenate them into synthetic images. While RDED achieves substantial performance improvements, it requires careful model selection and retraining of a classifier tailored to the specific dataset. Moreover, concatenated images often lack complete class-relevant features, especially for complex models with higher learning capacity. Model-Finetuned methods [4, 8] address distillation by pruning the DiT model [22] to generate synthetic datasets using regularization based on dataset representativeness and diversity. However, these methods rely on a diffusion model pre-trained on ImageNet-1K, requiring fine-tuning for different target datasets, which limits flexibility. Prototype-Finetuned methods [3,29] apply a pre-trained text-to-image Stable Diffusion (SD) [23] to generate synthetic datasets by finetuning original images in latent space. However, without constraints on distributional differences

between the LDM's pre-training data (e.g., LAION [25]) and the target dataset (e.g., ImageNet-1K), these methods can produce class-irrelevant or noisy images.

To overcome these limitations, we propose a novel dataset distillation paradigm that leverages diffusion model's prior to select the most informative image patches. Specifically, for each input image from the original dataset, we first predict the noise map using the pre-trained SD, with and without the label text prompt, calculating losses for each scenario. The differential loss quantifies representativeness, enabling us to crop patches most relevant to the label guided by the target dataset's data distribution. Additionally, we can effectively constrain the implicit data distribution learned by the diffusion model by using the pre-trained SD to identify class-pertinent regions.

To mitigate the inherent diversity of diffusion models, we apply clustering to capture the visual features most representative of each class. Specifically, we compute diffusion features for all patches inspired by recent work [31], allowing for efficient clustering and ranking of the cropped patches. Figure 1 illustrates a comparison between our method and previous diffusion-based generative distillation methods. Extensive experiments demonstrate that our approach achieves state-of-the-art (SOTA) performance on large-scale datasets, supporting an efficient, unrestricted, one-step distillation process.

Our contributions are summarized as follows:

- We introduce a novel dataset distillation framework that diverges from conventional methods by leveraging pre-trained diffusion models in an orthogonal way. Our approach uniquely addresses distributional discrepancies between the diffusion model and target dataset, offering a new perspective on improving dataset fidelity and applicability.
- Our proposed paradigm enables a highly efficient, one-step distillation process, eliminating the need for model training and fine-tuning, and avoiding repetitive distillation across class combinations or compression ratios.
- Extensive experiments demonstrate that our method consistently surpasses existing training-free methods across various settings.

2 Related Work

2.1 Dataset Distillation

Dataset distillation was initially regarded as a meta-learning problem [20,32,39]. It involves minimizing the loss function of the synthetic dataset through a model trained on this dataset. Since then, several approaches have been proposed to enhance the performance of dataset distillation.

Recent methods propose the decoupled distillation strategy and successfully extend the dataset distillation application to large-scale datasets. SRe^{2}L [33] leverages a pre-trained classifier to compute the cross-entropy loss of the synthetic dataset for updates, while RDED [30] calculates the confidence of randomly cropped patches from the original images with a pre-trained classifier and concatenates them together to form the synthetic dataset. Recent methods [2,38]

have introduced GAN as a parameterization technique. While with the development of generative models, the diffusion models are introduced to directly generate synthetic datasets, achieving promising performance [3,4,8,29,36]. However, existing diffusion-based methods are constrained by the implicit data distribution learned by diffusion models, leading to significant distributional bias in the synthetic dataset. In addition, aforementioned methods still fail to complete the distillation process in one pass for all settings. To address these issues, we utilize diffusion model to actively identify the most class-relevant regions in the original images, enabling an efficient dataset distillation process.

2.2 Diffusion Model

As a class of generative approaches, diffusion-based frameworks learn to progressively convert Gaussian-distributed noise vectors $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ into samples matching target distributions $\mathcal{X}$, demonstrating remarkable effectiveness across diverse applications [27, 40, 41]. These models employ a neural network $\epsilon_\theta(\mathbf{x}, t)$ parameterized by θ to estimate noise components, operating through sequential refinement steps denoted by a temporal parameter t. The denoising procedure typically involves iterative updates where the network processes corrupted inputs $\mathbf{x}$ at progressive noise levels determined by different timestap t.

The training protocol for latent diffusion models involves two core phases. Initially, an encoder network $E(\cdot)$ compresses input samples $\mathbf{x}$ into latent representations $\mathbf{z} = E(\mathbf{x})$. These latent codes undergo systematic corruption through a diffusion trajectory determined by uniformly sampled timestep t, implemented via the forward process formulation:

$$N_t(\mathbf{z}, \epsilon, t) = \sqrt{\bar{a}t}\mathbf{z} + (1 - \sqrt{\bar{a}t})\epsilon, \tag{1}$$

where N_t denotes the predicted noise at timesteps t, and the time-dependent attenuation factors $\sqrt{\bar{a}_t}$ govern the noise blending proportions. The denoising network ϵ_θ then learns to recover original signals, which is achieved through the optimization objective:

$$\mathcal{L}_t(\mathbf{z}, \epsilon) = |\epsilon_\theta(N_t(\mathbf{z}, \epsilon, t), t) - \epsilon|_2^2. \tag{2}$$

During the generation phase, novel samples are synthesized through a reverse diffusion process [13,28] that progressively refines Gaussian noise samples $\mathbf{z}_T$ over multiple iterations. For conditional generation tasks like text-to-image synthesis, the framework incorporates semantic conditioning signals c (e.g., text embeddings) as auxiliary inputs. This extends the denoiser architecture to $\epsilon_\theta(\mathbf{z}, t, c)$, modifying the training objective accordingly:

$$\mathcal{L}_t(\mathbf{z}, \epsilon, c) = |\epsilon_\theta(N_t(\mathbf{z}, \epsilon, t), t, c) - \epsilon|_2^2. \tag{3}$$

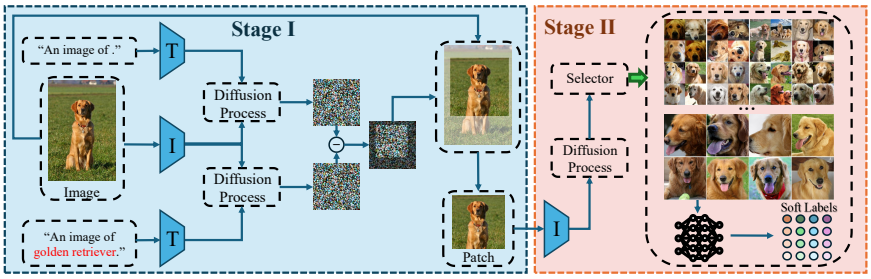


Fig. 2: (a): Stage I leverages image and corresponding differential class text prompts as inputs to the diffusion model to select the regions that most represent class-relevant features. (b): Stage II first computes DIFT of the patches and performs clustering to aggregate the most representative visual elements for each class. The evaluation process utilizes soft labels provided by a pre-trained teacher model. Rather than existing methods employing diffusion models, our proposed method maximizes the restoration of the original data distribution.

3 Method

In this section, we delve into the specifics of our method, aiming to address the limitations of previous methods. The overall framework of our method is shown in Figure 2.

3.1 Optimal Class-relevant Feature Selection

In recent years, several methods propose the utilization of pre-trained diffusion models to directly generate images in order to improve dataset distillation performance on large-scale datasets. However, these methods often face the challenge of distributional differences between the pre-training dataset and the target dataset. Model-finetuned methods [4,8] rely on fine-tuning a diffusion model pre-trained on ImageNet-1K, while Prototypes-finetuned methods [3,29] attempt to obtain representative samples through finetuning the original images in latent space. Both of them remain constrained by the distributional shifts introduced during the unconstrained denoising process.

To address the issue of data heterogeneity, we introduce a framework consisting of two stages that utilizing the diffusion model for localization rather than generation. As illustrated in Stage I of Figure 2, we first utilize diffusion models as zero-shot image-wise classifiers [19], which can be formulated as:

$$\begin{aligned} p(c_i|\mathbf{z}) &= \frac{\exp\{-\mathbb{E}_{\epsilon,t}[\epsilon_\theta(N_t(\mathbf{z},\epsilon,t),\ t,\ c_i)-\epsilon]\}}{\sum_j \exp\{-\mathbb{E}_{\epsilon,t}[\epsilon_\theta(N_t(\mathbf{z},\epsilon,t),\ t,\ c_j)-\epsilon]\}} \\ &= \frac{1}{\sum_j \exp\{\mathbb{E}_{\epsilon,t}[N_t(\mathbf{z},\epsilon,c_i)-N_t(\mathbf{z},\epsilon,c_j)]\}}. \end{aligned} \tag{4}$$

where $c_i \in [C]$ and $[C]$ represents the label space, $\mathbf{z}$ is the latent of input image $\mathbf{x}$. However, such an approach can only select the image with the highest

classification probability, lacking further constraints. Additionally, a larger class space can lead to significant computational overhead. Using the classifier-free guidance [14], we redefine the label space for a specific label c and rewrite Equation (4) as follow:

$$p(c|\mathbf{z}) = \frac{1}{1 + \exp\{\mathbb{E}_{\epsilon,t}[N_t(\mathbf{z}, \epsilon, c) - N_t(\mathbf{z}, \epsilon, \phi)]\}}, \tag{5}$$

where ϕ and c represents the classifier-free guidance and label text prompt respectively (i.e., "An image of ." and "An image of c"). We then simplify the computation as follow:

$$M(\mathbf{z}|c) = \mathbb{E}_{\epsilon,t}[N_t(\mathbf{z}, \epsilon, c) - N_t(\mathbf{z}, \epsilon, \phi)]. \tag{6}$$

Obviously Equation (5) and Equation (6) are monotonically equivalent. Based on the differential noise map $M(\mathbf{z}|c) \in \mathbb{R}^{H \times W \times 3}$, we could calculate the representative score $R(\mathbf{p}|c)$ within each patch $\mathbf{p}$ in original image. To enable computation on the original image, we first upsample the noise map from the latent space to the original image resolution via interpolation and take the channel-wise average to obtain $\hat{M}(\mathbf{x}|c) \in \mathbb{R}^{H \times W}$. We then apply an average pooling kernel to compute the score for each patch, as illustrated below:

$$R(\mathbf{p}|c) = \sum_{\mathbf{x}_i, \mathbf{x}_j \in \mathbf{p}} \hat{M}(\mathbf{x}|c)[i, j]. \tag{7}$$

Based on these representation scores, we can select the final distilled data and further reduce the complexity. The detailed workflow is depicted in Algorithm 1.

3.2 Visual Elements Aggregation

Although we have successfully selected the most representative patches, the overly complex data often introduce excessive randomness. To enhance the representativeness and reduce the randomness of the synthetic dataset, we perform an effective selecting strategy on the cropped patches as shown in the Stage II of Figure 2. With diffusion model, we select DIFT [31]for applying clustering, where noise at a specific time step is added to the latent of input image, and passed through a U-Net [24] with text condition to obtain the intermediate layer activations as features. On this basis, we use K-means [21] to cluster the selected patches.

To better leverage clustering information, we have designed a ranking rule: ***1) Intra-cluster.*** patches are ranked based on their distance to the centroid instead of corresponding $R(\mathbf{p}|c)$. ***2) Inter-cluster.*** clusters are ranked based on the median $R(\mathbf{p}|c)$ of the patches within each cluster. In practice, we fix the number of cluster centers at 32 and prioritize the selection of patches from the top 10 clusters under all compression settings to ensure both representativeness and diversity. Furthermore, when IPC ≤ 10, we follow the experimental setup of RDED by concatenating multiple patches into a single image. While for other settings, we treat each patch as an individual image.

Algorithm 1 Pseudocode of the proposed method

Input: $(\mathcal{T}, \mathcal{Y})$: Real dataset and corresponding label texts. $\mathcal{P}$: Selected patches. I: Pre-trained image encoder. T: Pre-trained text encoder. $\mathcal{U}$: Pre-trained time-conditional U-Net. T: Number of diffusion time-steps. C: Number of top cluster centers.

initialize $\mathcal{S} = \emptyset$
$Z = I(\mathcal{T}) \sim P_z$
for each class $Y \in \mathcal{Y}$ **do**
  $c = T(Y)$
  initialize $\mathcal{S}_Y, \mathcal{P}_Y = \emptyset$
  for each latent $\mathbf{z} \in Z$ **do**
    for i $\leftarrow 0$ to $N-1$ **do**
      $t \sim U(0.1, 0.7)$
      $N_i = N_t(\mathbf{z}, \epsilon, c) - N_t(\mathbf{z}, \epsilon, \phi)$
    end for
    $M(\mathbf{z}|c) = \frac{1}{T} \sum_i N_i$
    $\mathbf{p} = \arg\max_{\mathbf{p}} \sum_{\mathbf{x}_i, \mathbf{x}_j \in \mathbf{p}} \hat{M}(\mathbf{x}|c)[i, j])$
    $\mathcal{P}_Y \leftarrow \mathcal{P}_Y \cup \{\mathbf{p}\}$
  end for
  $cluster(\mathcal{U}(I(\mathcal{P}_Y), c, 160))$
  for i $\leftarrow 0$ to $C-1$ **do**
    $\mathcal{S}_Y \leftarrow \mathcal{S}_Y \cup top(\mathcal{P}_Y^i)$
  end for
  $\mathcal{S} \leftarrow \mathcal{S} \cup \mathcal{S}_Y$
end for

Output: $\mathcal{S}$: Distilled dataset.

4 Experiments

In this section, we demonstrate that our method outperforms existing approaches in both performance and computational efficiency. More experimental results and additional ablation studies are provided in the appendix.

4.1 Experimental Settings

Datasets and Architectures. To evaluate our proposed method on low-resolution datasets, we conduct experiments on CIFAR-10/100 [16] and TinyImageNet [17], with ConvNet [7] serving as the backbone. For high-resolution data, we perform evaluations on ImageNet-1K [5] and its widely used subsets, including ImageNet-100 [15], ImageNette and ImageWoof [1], utilizing ResNet-18 [10] as the backbone.

Baselines. we consider recent methods applicable to large-scale datasets as strong competitors. SRe^{2}L [33], by updating the synthetic dataset using the CE-loss of a pre-trained classifier and introducing soft label matching, was the first method to be feasibly applied to large-scale datasets. RDED [30] significantly improved computational efficiency and performance by evaluating randomly cropped patches with a pre-trained classifier and concatenating them

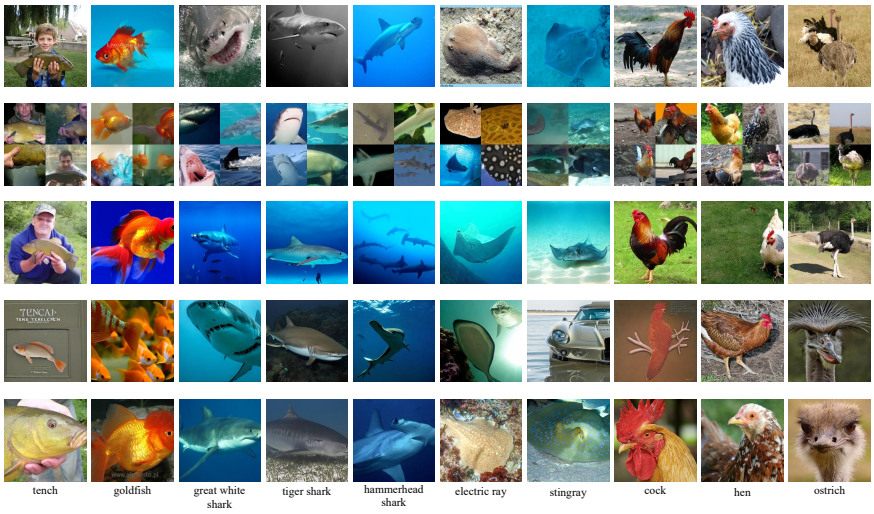


Fig. 3: Visualization comparison of images from the top ten classes of the ImageNet-1k dataset is presented (not cherry picked). From top to bottom, each row corresponds to the real dataset, RDED, Minimax, D^4M, and ours under IPC=50. Compared to existing methods, our method fully adheres to the distribution of the real dataset and demonstrates a sharper focus on class-specific features.

together to generate images. Minimax [8] and IGD [4] proposed fine-tuning DiT model pre-trained on ImageNet-1K by using the distribution of target dataset as a regularization to directly generate images. D^4M [29] and MGD^3 [3] extended the range of diffusion models to pre-trained text-to-image Stable Diffusion, utilizing the cluster centers of latent to generate synthetic datasets.

Evaluation Protocol. In line with previous work, we set IPC=10 and 50 for low-resolution datasets, and IPC=10, 50, and 100 for high-resolution datasets to enable more efficient learning. During the training process, we employ a fixed pre-trained classifiers to provide soft labels. Subsequently, five randomly initialized networks are trained on the distilled dataset, with the average performance of these networks evaluated. We use the same evaluation hyper-parameters with EDC [26].

Implementation Details. When calculating $M(\mathbf{z}|c)$ of images, we select a time-step range of [0.1, 0.7], whereas for the computation of DIFT, a time-step of 160 is employed. Across all IPC settings, we prioritize cropping the patches with resolution of 224×224 and selecting them from the top 10 ranked clusters after clustering. All the experiments were carried out using PyTorch on NVIDIA RTX-3090 GPUs.

Table 1: Performance comparison with methods applicable to large-scale datasets. Following the evaluation protocol of RDED, we use ResNet-18 as the teacher model for generating the dataset.

		ResNet-18		DiT-ImageNet			Stable Diffusion		
Dataset	IPC	SRe^{2}L	RDED	Minimax	IGD	D^{3}HR	D^{4}M	MGD^3	Ours
ImageNette	10	29.4 ± 3.0	61.4 ± 0.4	57.2 ± 0.8	58.3 ± 0.2	58.1 ± 0.2	57.4 ± 0.4	58.9 ± 0.4	**61.7 ± 0.4**
	50	40.9 ± 0.3	80.4 ± 0.4	84.4 ± 0.5	85.5 ± 0.3	85.7 ± 0.2	84.8 ± 0.2	85.2 ± 0.5	**86.4 ± 0.2**
	100	50.2 ± 0.4	89.6 ± 1.0	90.1 ± 0.6	91.6 ± 0.5	90.9 ± 0.2	90.4 ± 0.7	91.0 ± 0.2	**92.0 ± 0.3**
ImageWoof	10	20.2 ± 0.2	38.5 ± 2.1	38.1 ± 0.6	39.8 ± 0.1	39.3 ± 0.3	36.8 ± 1.2	38.2 ± 0.4	**44.5 ± 0.4**
	50	23.3 ± 0.3	68.5 ± 0.7	71.2 ± 0.6	72.8 ± 0.4	73.0 ± 0.2	71.4 ± 1.4	72.1 ± 0.5	**73.5 ± 1.4**
	100	37.6 ± 0.3	77.2 ± 0.6	77.4 ± 0.2	78.3 ± 0.4	78.7 ± 0.3	78.5 ± 0.3	77.9 ± 0.2	**79.6 ± 0.2**
ImageNet-100	10	9.5 ± 0.4	36.0 ± 0.3	31.6 ± 0.7	33.2 ± 0.3	34.0 ± 0.2	33.4 ± 0.2	34.2 ± 0.5	**36.8 ± 0.2**
	50	27.0 ± 0.4	61.6 ± 0.1	64.0 ± 0.5	64.2 ± 0.3	65.3 ± 0.2	62.7 ± 0.6	64.1 ± 0.3	**67.5 ± 0.3**
	100	39.4 ± 0.1	75.2 ± 0.3	76.4 ± 0.3	77.8 ± 0.3	77.2 ± 0.2	76.8 ± 0.5	76.7 ± 0.3	**78.6 ± 0.2**
ImageNet-1K	10	21.3 ± 0.6	42.0 ± 0.1	44.3 ± 0.3	45.5 ± 0.5	44.3 ± 0.3	27.9 ± 0.3	41.8 ± 0.2	**46.1 ± 0.3**
	50	46.8 ± 0.2	56.5 ± 0.1	58.6 ± 0.3	60.3 ± 0.4	59.4 ± 0.1	55.2 ± 0.3	57.2 ± 0.4	**61.0 ± 0.3**
	100	52.5 ± 0.5	59.5 ± 0.8	60.2 ± 0.3	61.4 ± 0.3	62.5 ± 0.0	59.3 ± 0.4	61.7 ± 0.5	**63.0 ± 0.2**

Table 2: Performance comparison on low-resolution datasets. To compare with information matching distillation methods, we use ConvNet-3 and ConvNet-4 as the backbone for CIFAR10/100 and TinyImagenet respectively. For decoupled distillation methods, we use ResNet-18 as the backbone. All the performance are evaluated on the same model architecture.

		ConvNet					ResNet-18		
Dataset	IPC	DSA	IDM	TESLA	RDED	Ours	SRe^{2}L	D^{4}M	Ours
CIFAR10	10	52.1 ± 0.5	58.6 ± 0.1	66.4 ± 0.8	**50.2 ± 0.3**	48.8 ± 0.4	29.3 ± 0.5	34.2 ± 0.3	**36.4 ± 0.2**
	50	60.6 ± 0.5	67.5 ± 0.1	72.6 ± 0.7	**68.4 ± 0.1**	66.2 ± 0.3	45.0 ± 0.7	60.9 ± 0.1	**61.0 ± 0.6**
CIFAR100	10	32.3 ± 0.3	45.1 ± 0.1	41.7 ± 0.3	**48.1 ± 0.3**	47.3 ± 0.4	27.0 ± 0.4	40.8 ± 0.5	**41.5 ± 0.4**
	50	42.8 ± 0.4	50.0 ± 0.2	47.9 ± 0.3	57.0 ± 0.1	**57.6 ± 0.2**	50.2 ± 0.4	62.3 ± 0.4	**63.8 ± 0.3**
TinyImageNet	10	-	21.9 ± 0.3	-	39.6 ± 0.1	**40.3 ± 0.2**	16.1 ± 0.2	35.1 ± 0.4	**40.2 ± 0.5**
	50	-	27.7 ± 0.3	-	47.6 ± 0.2	**49.2 ± 0.7**	41.1 ± 0.4	52.7 ± 0.4	**58.5 ± 0.2**

4.2 Performance Improvements

High-resolution Results. For high-resolution datasets, we first resize the smallest edge of images to 256 pixels while maintaining height-width ratio. The resized images are then fed into the diffusion model to calculate $M(\mathbf{z}|c)$ of the images and that of all patches with a 224×224 averaging pool kernel. For instance, when IPC = 100, we select 10 patches from the top ten clusters. If a cluster contains insufficient patches, we supplement the synthetic dataset by selecting the highest-scoring patches. As shown in Table 1, our proposed method demonstrates significant improvements across various datasets and IPC settings. In specific configurations, our approach exceeds baseline's performance by more than 10%, further validating the superiority of our method.

Low-resolution Results. For low-resolution datasets, we maintain the same setting as in RDED, utilizing the diffusion model to calculate the $M(\mathbf{z}|c)$ of full-

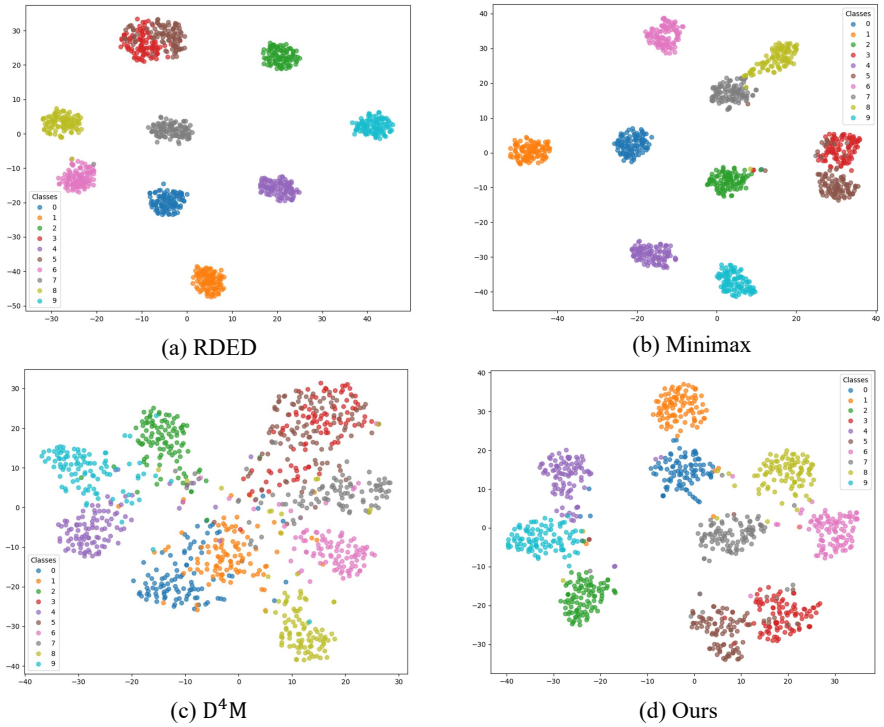


Fig. 4: Comparison of t-SNE visualizations of features extracted by a pre-trained ResNet-18 on ImageWoof under IPC=100.

image without cropping, followed by clustering, we then evaluate the synthetic datasets on the same architecture. As demonstrated in Table 2, the incorporation of the diffusion model yields performance on CIFAR-10 that is comparable to, or even exceeds, that of existing methods when applied within our proposed framework. On more complex datasets, such as CIFAR-100 and TinyImageNet, our approach outperforms all current optimization-based methods and achieves SOTA performance, surpassing all prior techniques.

4.3 Qualitative Analysis

To better understand the superiority of the datasets generated by our proposed method, we used t-SNE to visualize and analyze datasets produced by different methods. As shown in Figure 4, it is evident that the dataset generated by RDED [30] is overly concentrated around class centers because it originates from a pre-trained classifier. Additionally, since the pre-trained classifier cannot guarantee accurate classification results, the synthetic dataset from RDED suffers from class mixing issues. Similarly, the Minimax [8] and IGD [4], which uses the DiT pre-trained on ImageNet-1K to generate images, also results in datasets

Table 3: Synthetic dataset performance on ImageNet-1K with various teacher-student architectures under IPC=50. Our proposed method shows significant improvement with both CNNs and ViTs. "__" denotes the best performance of student network.

Teacher Network	Student Network						
	ResNet-18	ResNet-50	ResNet-101	MobileNet-V2	EfficientNet-B0	Swin-T	ViT-B
ResNet-18	<u>61.0</u>	63.1	65.7	53.6	60.4	<u>61.8</u>	<u>63.7</u>
ResNet-50	52.2	<u>64.8</u>	<u>66.7</u>	48.9	56.2	51.3	60.1
ResNet-101	50.1	63.7	66.5	46.1	55.5	42.8	50.5
MobileNet-V2	55.2	62.5	65.0	<u>55.3</u>	<u>60.5</u>	57.2	62.3
EfficientNet-B0	45.8	59.1	61.1	45.0	58.4	51.7	60.3
Swin-T	34.4	51.2	52.7	35.1	44.1	51.1	50.2
ViT-B	31.2	46.9	49.3	32.7	40.0	41.3	46.2

that are overly concentrated in class centers. This concentration can be detrimental to model learning when IPC increases. On the other hand, D^4M [29] and MGD^3 [3] generates images using SD, which is entirely unrelated to ImageNet-1K, resulting in datasets with excessive noise and outlier samples. In contrast to all the aforementioned methods, our approach achieves a good balance between diversity and representativeness by using SD for localization rather than generation. Furthermore, our method can still leverage diffusion models to generate images, which may lead to improved performance.

4.4 Cross-architecture Generalization

Following previous work, we evaluate the performance of the synthetic dataset using different teacher-student combinations. It can be observed that, when using similar network architectures as teacher-student pairs (e.g., both CNNs), shallower networks generally yield better performance as teacher models, while the larger networks show stronger learning ability and obtain the best performance when the size of synthetic datasets increases.

We further compare the performance of the synthetic dataset on extremely different network architectures, e.g., CNNs and ViTs. As a student network, the ViT-based networks leverages its global attention mechanism to attain the comparable performance with more realistic images. However, as a teacher network, ViT-based networks does not have strong ability to teach the student networks, yielding weak performance. Additionally, due to the low-pass filter characteristics of ViTs, effective learning becomes challenging when the sample size is limited and class-specific features are incomplete.

4.5 Training Efficiency

Distinct from all previous methods, our proposed method achieves a fully unrestricted distillation process and demonstrates SOTA performance. The limitations of existing dataset distillation methods are outlined as follows, and Table 4 represents a comprehensive comparison:

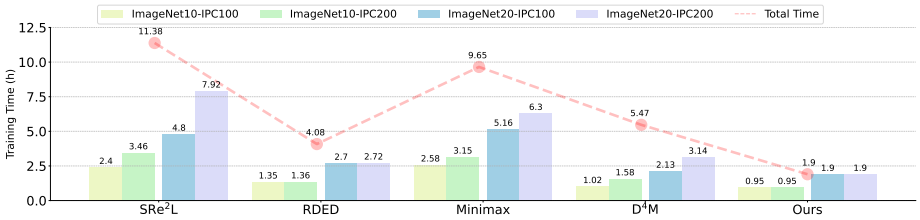


Fig. 5: The separate time cost of individual tasks and total time cost of finishing all the tasks. Our method achieves a constant time cost due to one-step distillation process, which eliminates the need for repeated distillation.

Table 4: We present an analysis of the generalization capabilities of various distillation methods under different settings, where ✓denote "Satisfactory".

	DATM	SRe^{2}L	Minimax	IGD	D^{4}M	MGD^3	RDED	Ours
Large scale	-	✓	✓	✓	✓	✓	✓	✓
Zero shot	-	-	-	-	✓	✓	-	✓
Class extension	-	-	-	-	✓	✓	✓	✓
IPC extension	-	✓	✓	✓	-	-	✓	✓

- **Large scale.** information-matching dataset distillation methods are impractical for large-scale datasets due to massive time cost.
- **Zero shot.** Relying on pre-trained classifiers require selecting an appropriate model architecture and training it for the target dataset. Minimax is also constrained by DiT pre-trained on ImageNet-1K.
- **Class extension.** Repeated distillation processes are required when dealing with different class-combination. For example, in ImageNet-1K, a 10-class classification task can result in C_{1000}^{10} possible combinations.
- **IPC extension.** Varying IPC settings necessitate multiple distillation. While D^{4}M addresses the class-combination issue, it necessitates recalculating different prototypes with different IPC settings.

For a more comprehensive comparison, we evaluate the total synthesis time of different methods under challenging compression settings. As shown in Figure 5, our one-step distillation method achieves significant efficiency across various tasks, demonstrating a significant improvement in computational efficiency compared to previous approaches.

4.6 Ablation Study

Clustering Strategy. To investigate the effectiveness of the two main stages of our proposed method, we conduct an ablation study to investigate whether clustering operations enhance the representative of synthetic datasets. Moreover,

Table 5: Performance Comparison on different data selecting strategys. Cluster-CLIP and Cluster-DIFT represent clustering with CLIP feature and DIFT respectively.

Ablation	IPC-10	IPC-50	IPC-100
	Dataset: ImageWoof		
w/o Cluster	41.8	70.4	79.1
w/ Cluster-CLIP	39.8	69.5	79.5
w/ Cluster-DIFT	**44.5(+2.7)**	**73.7(+3.3)**	**80.1(+1.0)**
	Dataset: ImageNet-1K		
w/o Cluster	43.6	58.1	62.0
w/ Cluster-CLIP	42.1	59.0	62.7
w/ Cluster-DIFT	**46.1(+2.5)**	**61.0(+1.9)**	**63.4(+1.4)**

we conduct experiments to explore the differences between using CLIP features and DIFT during the clustering process. The experimental results as shown in Table 5, demonstrates that the performance of the synthetic dataset improves further after incorporating the clustering operation. This indicates that reducing sample randomness enables the model to learn more representative information. Additionally, we found DIFT more effective for achieving clustering compared to CLIP feature.

Diffusion Time Intervals. To calculate the difference in predicted loss corresponding to the UNet when using different prompts (i.e., with and without the label text), we sample multiple diffusion time steps t as time conditions and average the results. As shown in Tab. 6, we selected different minimum and maximum boundaries for the time step range and evaluated the corresponding performance of the synthetic dataset. It can be observed that the dataset achieves the best performance when intermediate time steps are employed, while the inclusion of small time steps proves to be essential. This finding aligns with conclusions drawn in recent studies [19] [31], which highlight that early diffusion steps capture fine-grained details crucial for downstream tasks, whereas overly large time steps may introduce excessive noise that degrades representation quality. In practical applications, considering the generality of ImageNet-1K, we applied the same optimized range of diffusion time steps to all of its subsets (e.g., ImageWoof).

Table 6: Ablation study of the range of time steps on ImageNet-1K under IPC=50.

	End				
Start	0.1	0.3	0.5	0.7	0.9
0.1	-	58.9	59.3	**61.0**	59.7
0.3	-	-	58.8	58.9	59.6
0.5	-	-	58.1	58.6	59.0
0.7	-	-	-	-	57.9

Number of Real Images. Although performing the distillation operation once on the entire dataset offers the potential for one-step processing, the substantial time overhead makes this approach difficult to apply in practice. To

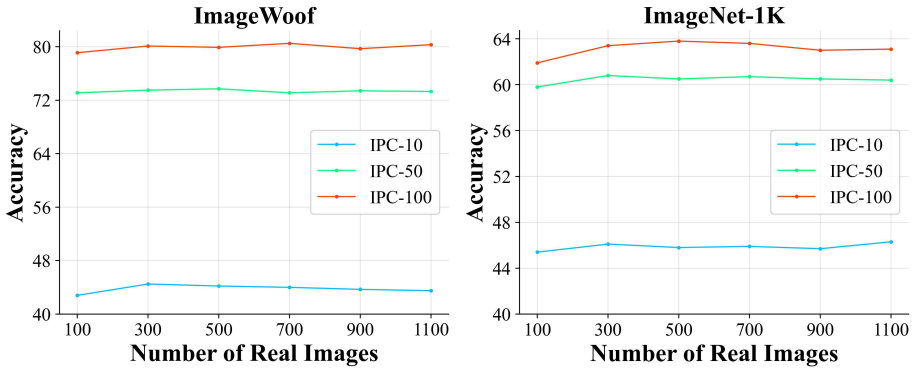


Fig. 6: Ablation study on the number of real images used.

address this issue, we explore the effect of the number of original images used for key patch extraction on performance. As shown in Figure 6, we randomly select images as a subset of the original dataset and evaluate the corresponding performance. Given that random selection is an unbiased data sampling method, our proposed achieves performance comparable to others, even when the number of original samples is limited. When 300 images are randomly selected from original dataset as input, the performance is nearly on par with the full dataset, resulting in a 4× speedup.

Time Steps on Computation of DIFT. An important component of our proposed framework involves calculating the DIFT [31] for all patches within the same class and using it as the feature for clustering. The process of calculating DIFT first requires performing a diffusion operation on the input patch at a specific time step. We conduct ablation experiments to investigate the impact of using different time steps. The experimental results are shown in Fig. 7. We hypothesis the phenomenon is because the earlier time steps of the diffusion model focus more on the low-frequency features of the image, which often provide more effective information for classification tasks. In practical applications, we choose $t = 160$ as the general setting.

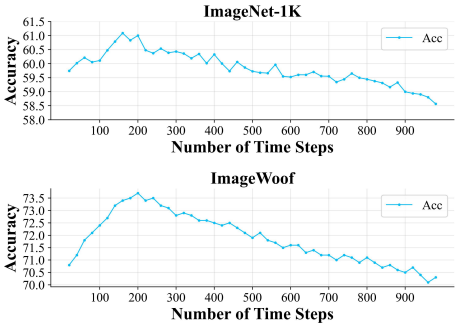


Fig. 7: Ablation study of time steps used to calculate the predicted noise under IPC=50.

5 Conclusion

We propose a novel approach that operates orthogonally to existing diffusion-based generative dataset distillation methods, addressing the challenge of distribution mismatch between pre-trained diffusion models and target datasets. Through a two-stage framework, our method achieves a highly efficient, unrestricted one-step distillation process, eliminating the need for repeated distillations under varied settings. Extensive experiments demonstrate that our method significantly improves performance on large-scale datasets and deeper models, while achieving competitive results on low-resolution datasets. Additionally, our work provides a fresh perspective on leveraging diffusion models for dataset distillation.

Acknowlegement. This work is supported in part by the National Science and Technology Major Project under grant 2025ZD1601000, National Natural Science Foundation of China under grant 62301189, 62576122,62571298, Guangdong Basic and Applied Basic Research Foundation under grant 2026A1515011139.

References

1. Cazenavette, G., Wang, T., Torralba, A., Efros, A.A., Zhu, J.Y.: Dataset distillation by matching training trajectories. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4750–4759 (2022)
2. Cazenavette, G., Wang, T., Torralba, A., Efros, A.A., Zhu, J.Y.: Generalizing dataset distillation via deep generative prior. arXiv preprint arXiv:2305.01649 (2023)
3. Chan-Santiago, J.A., Tirupattur, P., Nayak, G.K., Liu, G., Shah, M.: Mgd3: Mode-guided dataset distillation using diffusion models. arXiv preprint arXiv:2505.18963 (2025)
4. Chen, M., Du, J., Huang, B., Wang, Y., Zhang, X., Wang, W.: Influence-guided diffusion for dataset distillation. In: The Thirteenth International Conference on Learning Representations (2025)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
6. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
7. Gidaris, S., Komodakis, N.: Dynamic few-shot visual learning without forgetting. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4367–4375 (2018)
8. Gu, J., Vahidian, S., Kungurtsev, V., Wang, H., Jiang, W., You, Y., Chen, Y.: Efficient dataset distillation via minimax diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15793–15803 (2024)
9. Gu, X., Zhong, X., Wei, Z., Zhou, Y., Sun, S., Chen, B., Wang, H., Luo, Y.: Temporal saliency-guided distillation: A scalable framework for distilling video datasets. arXiv preprint arXiv:2505.20694 (2025)

10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
11. He, Y., Xiao, L., Zhou, J.T.: You only condense once: Two rules for pruning condensed datasets. Advances in Neural Information Processing Systems **36** (2024)
12. He, Y., Xiao, L., Zhou, J.T., Tsang, I.: Multisize dataset condensation. arXiv preprint arXiv:2403.06075 (2024)
13. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. NeurIPS (2020)
14. Ho, J., Salimans, T.: Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598 (2022)
15. Kim, J.H., Kim, J., Oh, S.J., Yun, S., Song, H., Jeong, J., Ha, J.W., Song, H.O.: Dataset condensation via efficient synthetic-data parameterization. In: International Conference on Machine Learning. pp. 11102–11118. PMLR (2022)
16. Krizhevsky, A.: Learning multiple layers of features from tiny images. Master's thesis, University of Tront (2009)
17. Le, Y., Yang, X.: Tiny imagenet visual recognition challenge. CS 231N **7**(7), 3 (2015)
18. Lei, S., Tao, D.: A comprehensive survey to dataset distillation. arXiv preprint arXiv:2301.05603 (2023)
19. Li, A.C., Prabhudesai, M., Duggal, S., Brown, E., Pathak, D.: Your diffusion model is secretly a zero-shot classifier. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2206–2217 (2023)
20. Li, Z., Zhong, X., Khaki, S., Liang, Z., Zhou, Y., Shi, M., Wang, Z., Zhao, X., Zhao, W., Qin, Z., et al.: Dd-ranking: Rethinking the evaluation of dataset distillation. arXiv preprint arXiv:2505.13300 (2025)
21. Lloyd, S.: Least squares quantization in pcm. IEEE transactions on information theory **28**(2), 129–137 (1982)
22. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4195–4205 (2023)
23. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
24. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18. pp. 234–241. Springer (2015)
25. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al.: Laion-5b: An open large-scale dataset for training next generation image-text models. Advances in Neural Information Processing Systems **35**, 25278–25294 (2022)
26. Shao, S., Zhou, Z., Chen, H., Shen, Z.: Elucidating the design space of dataset condensation. Advances in neural information processing systems **37**, 99161–99201 (2024)
27. Siglidis, I., Holynski, A., Efros, A.A., Aubry, M., Ginosar, S.: Diffusion models as data mining tools. In: European Conference on Computer Vision. pp. 393–409. Springer (2024)
28. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. ICLR (2021)

29. Su, D., Hou, J., Gao, W., Tian, Y., Tang, B.: Dˆ 4: Dataset distillation via disentangled diffusion model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5809–5818 (2024)
30. Sun, P., Shi, B., Yu, D., Lin, T.: On the diversity and realism of distilled dataset: An efficient dataset distillation paradigm. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9390–9399 (2024)
31. Tang, L., Jia, M., Wang, Q., Phoo, C.P., Hariharan, B.: Emergent correspondence from image diffusion. Advances in Neural Information Processing Systems **36**, 1363–1389 (2023)
32. Wang, T., Zhu, J.Y., Torralba, A., Efros, A.A.: Dataset distillation. arXiv preprint arXiv:1811.10959 (2018)
33. Yin, Z., Xing, E., Shen, Z.: Squeeze, recover and relabel: Dataset condensation at imagenet scale from a new perspective. Advances in Neural Information Processing Systems **36** (2024)
34. Zhao, B., Bilen, H.: Dataset condensation with differentiable siamese augmentation. In: International Conference on Machine Learning. pp. 12674–12685. PMLR (2021)
35. Zhao, B., Bilen, H.: Dataset condensation with distribution matching. arXiv preprint arXiv:2110.04181 (2021)
36. Zhao, L., Wu, Y., Jiang, X., Gu, J., Wang, Y., Xu, X., Zhao, P., Lin, X.: Taming diffusion for dataset distillation with high representativeness. arXiv preprint arXiv:2505.18399 (2025)
37. Zhong, X., Chen, B., Fang, H., Gu, X., Xia, S.T., Yang, E.H.: Going beyond feature similarity: effective dataset distillation based on class-aware conditional mutual information. In: International Conference on Learning Representations. vol. 2025, pp. 36598–36615 (2025)
38. Zhong, X., Fang, H., Chen, B., Gu, X., Dai, T., Qiu, M., Xia, S.T.: Hierarchical features matter: A deep exploration of gan priors for improved dataset distillation. arXiv preprint arXiv:2406.05704 (2024)
39. Zhong, X., Sun, S., Gu, X., Zhu, C., Chen, B., Wang, Y.: Rectified decoupled dataset distillation: A closer look for fair and comprehensive evaluation. arXiv preprint arXiv:2509.19743 (2025)
40. Zhu, C., Li, K., Ma, Y., He, C., Xiu, L.: Multibooth: Towards generating all your concepts in an image from text. arXiv preprint arXiv:2404.14239 (2024)
41. Zhu, C., Li, K., Ma, Y., Tang, L., Fang, C., Chen, C., Chen, Q., Li, X.: Instantswap: Fast customized concept swapping across sharp shape differences. arXiv preprint arXiv:2412.01197 (2024)

A More Experimental Results

A.1 Hard Label Performance

To further investigate the quality of synthetic datasets generated by different distillation methods, we conduct an experiment with one-hot labels.

As shown in Tab. 7, SRe^{2}L exhibit extreme poor performance, Due to the lack of diversity of generated images caused by using only cross-entropy loss. For Minimax and RDED, the synthetic datasets also exhibit concentration of class centers which means the loss of diversity, leading a performance degradation especially on large IPC settings. Despite D^{4}M successfully enhance diversity of synthetic datasets by using a SD trained on LAION, the misalignment phenomenon observed between images and corresponding class labels, leading to a significant performance drop when using hard label. In contrast, our method achieve a balance between representative and diversity which has also been verified using t-SNE visualization, thus a strong performance achieved using only hard label.

Table 7: Hard label performance on ImageNet-1K with different methods.

IPC	Random	SRe^{2}L	Minimax	D^{4}M	RDED	Ours
1	1.1±0.3	0.2 ±0.1	1.3 ± 0.7	0.7 ± 0.4	1.1 ± 0.3	**1.7 ± 0.3**
10	7.3 ±0.2	1.4 ± 0.2	8.2 ± 0.4	4.6 ± 0.5	**11.4 ± 0.4**	10.5 ± 0.3
50	30.9 ±0.1	3.2 ± 0.3	31.3 ± 0.4	17.6 ± 0.2	30.1 ± 0.3	**33.2 ± 0.1**
100	41.0 ±0.3	4.1 ± 0.3	40.5 ± 0.2	27.3 ± 0.5	38.4 ± 0.2	**42.1 ± 0.2**

A.2 Generalization Ability

To better investigate the generalization capability of our proposed method under different settings, we evaluated its performance using various teacher-student

Table 8: Synthetic dataset performance on ImageNet-1K with various teacher-student architectures under IPC=10.

Teacher Network	Student Network						
	ResNet-18	ResNet-50	ResNet-101	MobileNet-V2	EfficientNet-B0	Swin-T	ViT-B
ResNet-18	<u>46.1</u>	<u>50.1</u>	<u>51.2</u>	36.1	<u>46.1</u>	<u>31.2</u>	<u>22.3</u>
ResNet-50	34.6	44.0	42.2	27.8	38.1	27.4	19.8
ResNet-101	33.2	41.4	42.5	28.2	37.0	26.1	17.0
MobileNet-V2	40.1	48.3	46.7	<u>37.5</u>	46.0	30.1	20.5
EfficientNet-B0	29.8	38.4	38.7	27.4	38.9	30.3	21.3
Swin-T	18.9	30.2	31.4	21.6	31.7	26.7	16.0
ViT-B	16.8	27.8	28.9	19.2	28.9	23.6	15.0

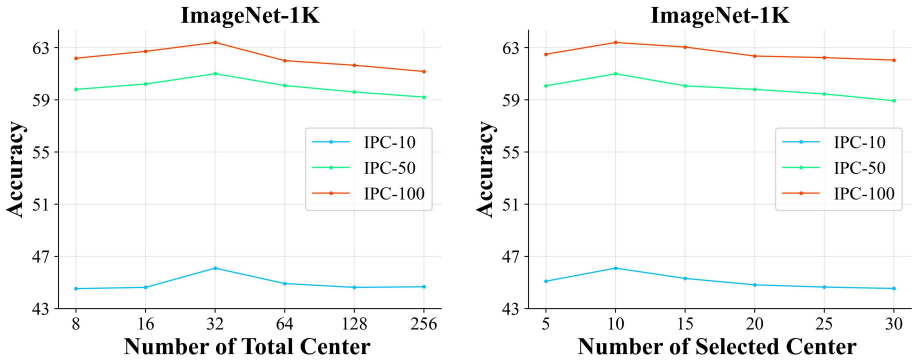


Fig. 8: Ablation study of the number of cluster center M and the number of selected cluster N on ImageNet-1K.

model pairs at IPC=10. Unlike the IPC=50 setting, each image generated is composed of four patches.

As shown in Tab. 8, our proposed method still demonstrates strong performance on CNNs. Additionally, it can be observed that ResNet-18 is the most suitable architecture to be used as a teacher model when the sample size is smaller. However, for ViTs, due to their low-pass filter characteristics, effective learning becomes challenging when the sample size is limited and class-specific features are incomplete. The best performance achieved when using DiT as a student model is only 50% of that of the CNN student model under the same settings.

B Additional Ablation Studies

B.1 Number of Cluster Centers

To enhance the representativeness of the synthetic dataset, we performed clustering on all patches to extract the most class-relevant visual elements. To gain deeper insights into the impact of the clustering operation on the performance of the synthetic dataset, we conduct additional ablation experiments focusing on the number of cluster centers. Specifically, we evaluate the performance of the synthetic dataset using different numbers of cluster centers M, as well as the number N of top clusters used for extracting patches after clustering. In practical applications, when IPC ≤ 10, the number of patches provided by each cluster is 4*IPC // N, otherwise the number is IPC // N. The remaining patches are selected sequentially from the unused ones based on $R(p|c)$ of each patch p in descending order.

The experimental results are shown in Fig. 8, indicating that when N is fixed, both too few or too many clusters lead to a failure in balancing diversity and representativeness in the synthetic dataset, resulting in a decline in performance.

A similar trend was observed in the ablation experiment where M was fixed and N varied.

B.2 Impact of Resizing

When using diffusion to process images, the image size often has a certain degree of impact. On the other hand, when calculating $R(p|c)$ for each patch, we used a fixed-size average pooling kernel to obtain the mean value of the patch. Therefore, we designed an ablation experiment to explore whether different input image sizes and patch sizes significantly affect the performance of the synthetic dataset. In practical applications, we proportionally scale the original image such that its size of shortest edge matches the selected size. For different patch sizes, under the setting of IPC=10, we scale all patches to 112×112 in order to concatenate 4 patches into a 224×224 image. Under the setting of IPC=50, we scale all patches to 224×224.



(a) 112x112

(b) 224x224

Fig. 9: Visualizations of the patches with different cropped size within class "Golden Retriever".

The experimental results, as shown in Tab. 9, indicate that when the original image size closely matches the standard output size of SD v1.5, better performance are achieved. Additionally, when the size of the cropped patches is closer to the scaled original image size, the image tends to exhibit more complete class-relevant features, leading to better performance. Consider the balance between performance and efficiency, We choose 256x256 as the size of the real images, and 224×224 as the size of the patches. Comparison of visualization with different crop sizes when using 256x256 as the size of the real images are shown in Fig. 9, An excessive size difference between patches and the original image can cause the generated dataset to overly focus on low-frequency features (e.g., noses of dogs), leading to performance degradation.

C Visualizations

To facilitate a more intuitive understanding of our methodology, we present the complete synthetic datasets for ImageWoof and ImageNette at IPC=10 as illustrated in Figs. 10 and 11, as well as the partial synthetic dataset at IPC=50,

Table 9: Ablation study of different size of real images and patches on ImageWoof

Patch Size	IPC-10	IPC-50	IPC-100
	Image Size: 128×128		
112×112	42.6	70.2	77.8
	Image Size: 256×256		
112×112	40.4	67.4	76.8
224×224	44.5	73.7	80.1
	Image Size: 512×512		
112×112	38.6	65.3	71.4
224×224	43.1	69.5	78.4
448×448	45.1	73.2	81.0

depicted in Figs. 12 and 13. Furthermore, we have visualized all images belonging to the “Golden Retriever" class at IPC=100, as shown in Fig. 14.

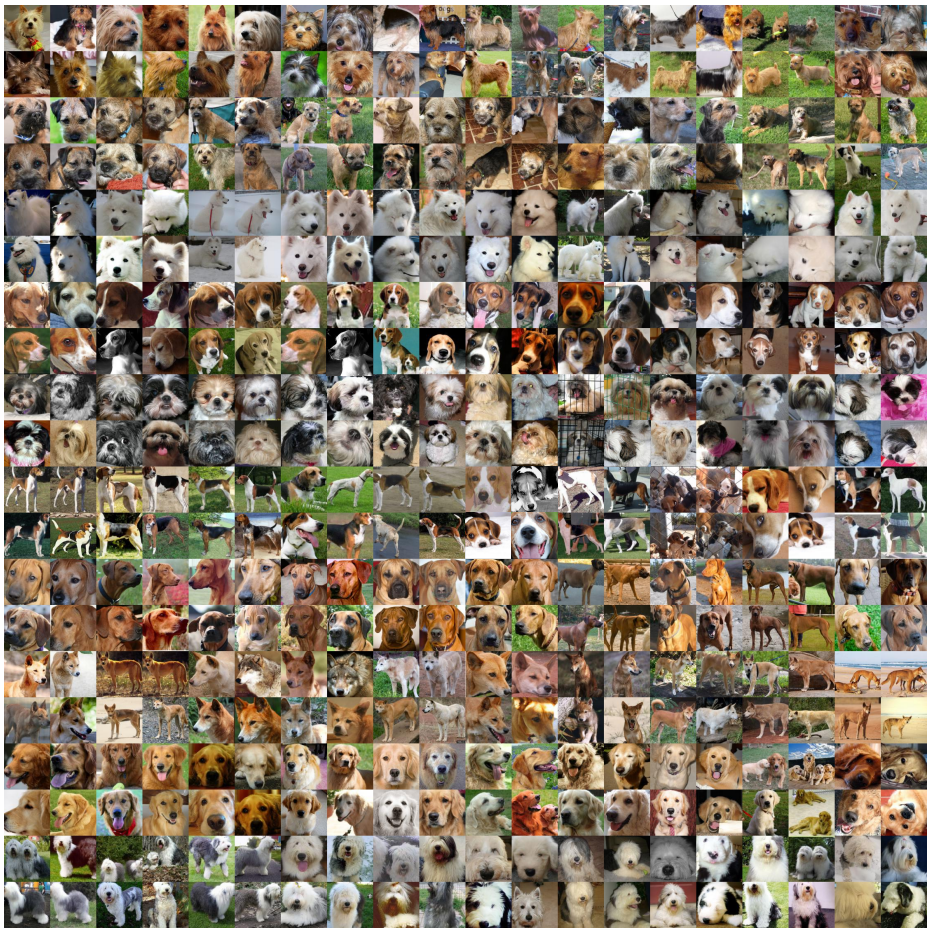


Fig. 10: Visualization of ImageWoof under IPC=10.

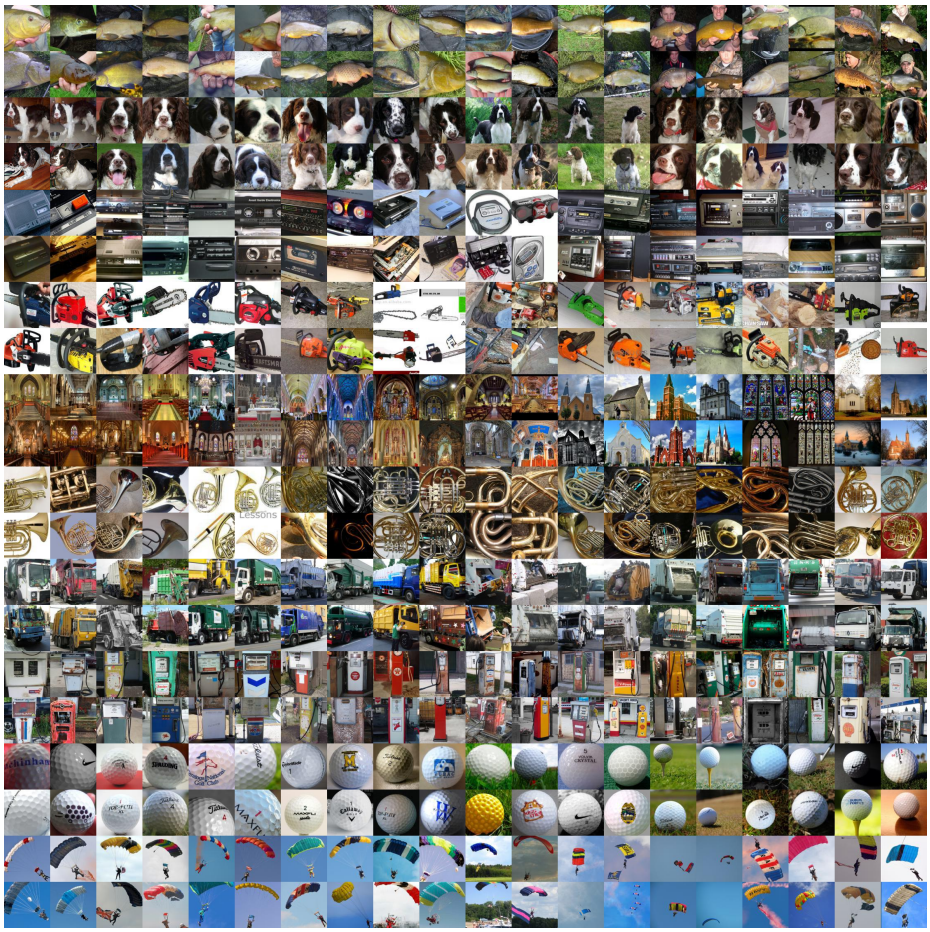


Fig. 11: Visualization of ImageNette under IPC=10.



Fig. 12: Partial visualization of ImageWoof under IPC=50.



Fig. 13: Partial visualization of ImageNette under IPC=50.



Fig. 14: Visualization of class "Golden Retriever" under IPC=100.