Diffusion-Augmented Coreset Expansion for Scalable Dataset Distillation

Ali Abbasi[1*] Shima Imani[2†] Chenyang An[3] Gayathri Mahalingam[2]
Harsh Shrivastava[2] Maurice Diesendruck[2†] Hamed Pirsiavash[4]
Pramod Sharma[2‡] Soheil Kolouri[1‡]
[1]Vanderbilt University [2]Microsoft Research [3]University of California, San Diego
[4]University of California, Davis
{ali.abbasi, soheil.kolouri}@vanderbilt.edu
c5an@ucsd.edu
{gmahalingam, hshrivastava, pramod.sharma}@microsoft.com
moglobal@gmail.com, shimaimani@meta.com

Abstract

With the rapid scaling of neural networks, data storage and communication demands have intensified. Dataset distillation has emerged as a promising solution, condensing information from extensive datasets into a compact set of synthetic samples by solving a bilevel optimization problem. However, current methods face challenges in computational efficiency, particularly with high-resolution data and complex architectures. Recently, knowledge-distillation-based dataset condensation approaches have made this process more computationally feasible. Yet, with the recent developments of generative foundation models, there is now an opportunity to achieve even greater compression, enhance the quality of distilled data, and introduce valuable diversity into the data representation. In this work, we propose a two-stage solution. First, we compress the dataset by selecting only the most informative patches to form a coreset. Next, we leverage a generative foundation model to dynamically expand this compressed set in real-time—enhancing the resolution of these patches and introducing controlled variability to the coreset. Our extensive experiments demonstrate the robustness and efficiency of our approach across a range of dataset distillation benchmarks. We demonstrate a significant improvement of over 10% compared to the state-of-the-art on several large-scale dataset distillation benchmarks.

1. Introduction

With the rapid advancement of deep learning, the scale of neural networks and the datasets required to train them have expanded dramatically, introducing significant computational challenges. One promising approach to mitigate these demands is to explore the potential of "small data," a research direction introduced by Wang et al. [34] and known as dataset distillation. Dataset distillation focuses on synthesizing a compact yet highly informative dataset from the original large-scale data, allowing models trained on this smaller set to achieve performance comparable to those trained on the full dataset [37]. By reducing training overhead, storage, and communication requirements while preserving the essential knowledge of the larger dataset, dataset distillation offers transformative potential across multiple areas of machine learning research. Its impact is particularly significant in applications that 1) require repeated training over large datasets, as seen in neural architecture search [27], 2) depend on constrained memory storage, such as memory replay in continual learning [21], and 3) involve knowledge sharing and communication across distributed machine learning agents, such as in federated learning [35].

Dataset distillation is classically formulated as a bilevel optimization problem, where the inner loop trains a model on a synthesized dataset, and the outer loop adjusts this synthetic dataset to maximize the model's performance on the original large-scale dataset. However, this approach presents two significant challenges: 1) it is computationally and memory intensive, as the outer optimization requires backpropagation through the entire unrolled computation graph of the model's training process in the inner loop; and 2) it often leads to synthetic images with spurious, non-realistic features due to overfitting to the specific architecture used during optimization, which limits generalization across ar-

* Work was partly done during an internship at Microsoft Research.
† Contributed to the project while working at Microsoft Research.
‡ Equal contribution.

chitectures [2]. To address the former challenge and scale up computation, researchers have proposed methods such as gradient matching [41], which aligns the gradients of synthetic and original data to improve scalability, and training trajectory matching [1], which matches training trajectories between models trained on synthetic and original datasets to enhance distillation efficiency. To address the latter problem, recent studies emphasize the importance of realism in distilled data for achieving cross-architecture generalizability [2, 24, 28, 36], showing that incorporating generative priors and enhancing the diversity and realism of synthetic datasets can significantly improve the generalization capabilities of models trained on distilled data.

Recently, Sun et al. [28] introduced the Realistic, Diverse, and Efficient Dataset Distillation (RDED) method, an optimization-free approach that achieves high-resolution and large-scale image dataset distillation by emphasizing the realism and diversity of the distilled images. RDED selects a diverse set of informative patches directly cropped from the original data and combines these patches into new images to form the synthetic dataset. To guide this selection, RDED uses a "teacher" model trained on the large-scale dataset to identify informative patches and assign soft labels to them. A "student" model is then trained on these patches and their corresponding soft labels, effectively employing knowledge distillation for dataset distillation.

In RDED, we observe a trade-off between patch diversity and realism for a fixed compression budget, i.e., images per class (IPC). Increasing diversity requires reducing the size of patches to fit more patches within the limited pixel space. However, decreasing patch sizes also reduces their realism, as downsampling acts as a low-pass filter, causing a loss of fine-grained details. In this context, we pose two questions: 1) Is it possible to pack more patches into a finite pixel space without sacrificing realism? and 2) Can we enhance diversity within a fixed number of patches? Following the RDED framework [28], to increase the number of patches without sacrificing realism, super-resolution techniques can be used to enhance low-resolution patches back to high-resolution quality. Additionally, diversity within a fixed number of patches can be achieved through realistic augmentations that preserve the natural image manifold. We show that modern Latent Diffusion Models (LDMs) provide both these capabilities, enabling high-quality super-resolution and naturalistic diversity enhancements. We show that enhancing the realism and diversity of the distilled dataset using LDMs results in a performance boost across various dataset distillation benchmarks.

As fast LDMs [20, 23] become commonplace in ML workflows, on-the-fly diffusion augmentation grows practical, and consistent teacher-student processing through a shared LDM becomes a reasonable assumption.

Our experiments across multiple datasets and model architectures demonstrate that distilling a dataset into a small set of images (e.g., an image per category) and training a student model on this distilled set results in higher accuracy than state-of-the-art dataset distillation methods. On the ImageNette [13] dataset using a ResNet-18 architecture, our approach achieves 51.4% accuracy, while RDED [28], a recent comparable baseline, reaches only 35.8% accuracy.

Contributions: (i) fast LDMs for on-the-fly coreset expansion in dataset distillation; (ii) knowledge distillation paired with generative models for dataset distillation; (iii) state-of-the-art performance on large-scale dataset distillation benchmarks.

2. Related Works

Since its introduction by Wang et al. [34], numerous variations of dataset distillation have been developed. Below, we review some of these methods as well as other related works to our proposed framework.

Bi-level optimization provides a natural framework for dataset distillation [34]. However, this approach is generally intractable due to the significant computational and memory demands required to backpropagate through the unrolled computational graph of the inner model optimization. Various methods have been proposed to ameliorate this issue by introducing surrogate objective functions for the outer optimization. These include methods that match gradients [15, 41], extracted features [33] and their distributions [40], and training trajectories [1, 7].

Core-set selection offers a natural approach to dataset distillation by selecting the most "valuable" subset of the training data rather than synthesizing a condensed version. Core-set-based methods vary primarily in the difficulty-based metrics they use to evaluate sample importance. For example, sample scores such as Gradient Normed (GraNd) and Error L2 Norm (EL2N) were introduced in [19] to guide core-set selection. Meanwhile, the forgetting score, initially proposed in [29], has recently been employed to perform dataset distillation progressively [3]. Recent works extend the concept of core-set selection by selecting a subset of pixels, tokens, or patches within chosen images, achieving further dataset compression [28, 43]. For instance, Sun et al. [28] select important image patches identified by a teacher model, while [43] use a subset of image tokens or patches and apply Masked Auto-Encoders [11] to reconstruct the missing patches, resulting in dataset distillation conditioned on a generative model. In our work, we use important patches similar to [28] but leverage generative modeling, and more precisely LDMs, to increase diversity and maximize compression, similar to [43].

Realism priors have been explored in several studies to enhance the realism of distilled data. For example, Cazenavette et al. [1] show that overfitting to a specific architecture often stems from optimizing in the pixel space, which re-

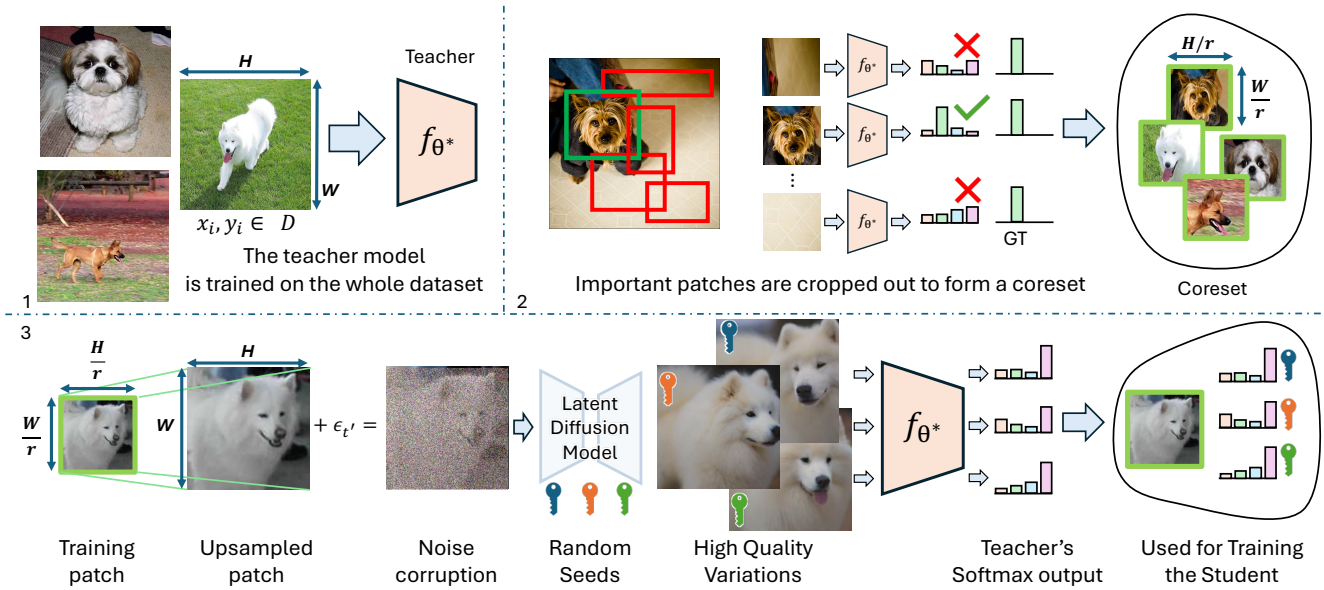

Figure 1. Proposed framework: An image dataset $\mathcal{D}$ trains a teacher model, which identifies key patches via uncertainty signals [28]. These patches are upsampled, noise-corrupted, and processed through a multi-step diffusion model for super-resolution and variation. The teacher provides soft labels for each generated image. The student reconstructs high-resolution images using patches and random seeds, regressing over the teacher's soft labels. Like geometric augmentations, this process occurs on the fly.

duces realism. They address this by using generative models—specifically, Generative Adversarial Networks (GANs), to perform optimization in the latent space, producing a more realistic distilled dataset leading to better cross-architecture generalization. Other approaches use a trained teacher model and align the feature statistics of synthetic distilled data with those of a larger dataset [24, 36], which implicitly improves image realism. In our work, we leverage generative priors to enhance both the realism and diversity of the distilled dataset.

Diffusion models have recently gained prominence as powerful tools for data augmentation and generation across various learning tasks [14, 22, 25, 31, 39]. For example, Zhang et al. [39] combine diffusion models with MAEs to expand small-scale datasets by generating new, informative, and diverse images, effectively creating realism-aware augmentations of limited datasets. DiffuseMix [14] utilizes diffusion models and introduces a unique approach that blends real and generated images, producing hybrid augmentations. However, due to the slow generation speed of diffusion models, these augmentations are often pre-generated and cached, leading to high memory demands. With growing interest in diffusion models and advances in fast sampling techniques, such as SDXL-Turbo [23], it is now feasible to generate on-the-fly augmentations during model training. In our work, we leverage SDXL-Turbo to super-resolve and augment our mined, important patches.

3. Method

Our proposed method for dataset distillation comprises three main steps. In Step 1, a teacher model is trained on the full training dataset. In Step 2, a compact coreset of important image patches is selected. In Step 3, the selected image patches are first upsampled and then noise-corrupted using different fixed random seeds. Each seed generates a distinct high-quality variation of the low-resolution patch using a Latent Diffusion Model (LDM). Multiple such variations are generated for each patch using different seeds. The high-quality images are then processed by the teacher model to obtain the softmax outputs of its classifier. Finally, the low-quality important patches, random seeds, and their corresponding soft labels are transferred to the student model. Figure 1 demonstrates these steps.

Upon receiving the distilled data from the teacher, the student replicates the upsampling, noise corruption, and LDM denoising steps using the provided low-quality patches and random seeds to generate the same high-quality variations as produced by the teacher. The student then trains on the teacher's soft labels for the generated images.

3.1. Coreset Selection

We follow the methodology of [28] for forming the coreset of important patches. Given $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, where $x_i \in \mathbb{R}^{H \times W \times 3}$ are the images and $y_i \in \mathbb{R}^K$ are the corresponding class labels, we first train a teacher model $f_\theta : \mathbb{R}^{H \times W \times 3} \rightarrow \mathbb{R}^K$ parameterized by θ on $\mathcal{D}$. Then, for a given image x_i from the dataset, we generate P random crops and resize them to be $\lfloor \frac{H}{r} \rfloor \times \lfloor \frac{W}{r} \rfloor$, where $r > 1$ is a scalar indicating

the patch-to-image compression ratio. Denoting x_i^j as the j-th random patch from x_i, we deliberately use a smaller patch size compared to the full dimensions of x_i to compress the information, i.e., $x_i^j \in \mathbb{R}^{\lfloor H/r \rfloor \times \lfloor W/r \rfloor \times 3}$. To construct our coreset, we first select the most informative patch from each image. This is achieved by choosing the patch x_i^j that minimizes the cross-entropy loss:

$$x_i^* = \arg\min_j \text{CE}(f_\theta(x_i^j), y_i), \tag{1}$$

where $j = 1, 2, \ldots, P$, and $\text{CE}(\cdot, \cdot)$ denotes the cross entropy loss. Next, to create the coreset under a fixed Images Per Class (IPC) memory budget, we form the coreset by choosing P patches with the lowest cross entropy loss. At the end of this step, for each class c, we will have a total of $IPC \times r^2$ patches, satisfying the memory constraint. Figure 2 shows the selected coreset for the ImageNette dataset [13] for IPC=1. We note that increasing r allows us to store a greater number of important patches for a fixed IPC budget, thereby enhancing diversity. However, this comes at the cost of reduced realism due to the loss in resolution. Finally, we acknowledge that selecting patches based on minimum cross-entropy does not inherently ensure diversity. However, our results demonstrate that the diffusion model effectively compensates for any potential lack of diversity among the selected important patches. Next, we will describe our method for increasing realism despite increasing r.

3.2. Coreset Augmentation and Super Resolution

Data augmentation has long been a cornerstone for introducing variations into training data. In vision applications, simple and efficient geometric transformations such as rotations, flips, and noise additions are commonly used to artificially increase the dataset size. These augmentations are computationally inexpensive and can be performed on the fly. With recent advancements in Latent Diffusion Models (LDMs) [23], the development of few-step LDMs has significantly reduced the sampling latency traditionally associated with diffusion models. In this work, we leverage this advancement to perform realistic augmentations on important patches on the fly. Realistic variations are generated dynamically during the student model's batch processing and discarded once the batch is processed, ensuring compliance with memory storage constraints on the client side.

3.2.1. Diffusion Preliminary

Latent Diffusion Models (LDMs) consist of an autoencoder and a UNet denoiser. The autoencoder is an encoder-decoder architecture that is initially trained separately to map the image from pixel space to a lower-dimensional latent space and then reconstruct it back to the original image space with minimal reconstruction error. Once the autoencoder is trained, all noise corruptions and denoising are performed in the latent space. Let $z_0 = \text{Encode}(x)$ be the latent representation of

Figure 2. The extracted coreset for IPC=1 from ImageNette.

image x, and let $t \in \{0, 1, 2, \ldots, T\}$ represent an arbitrary noising step. For training an LDM, t is uniformly sampled from the set of steps, and Gaussian noise is added to z_0 in proportion to t. The noisy latent representation is given by:

$$z_t = \sqrt{\bar{\alpha}_t} z_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon,$$

where $\sqrt{\bar{\alpha}_t}$ is the data-noise interpolation coefficient and $\epsilon \sim \mathcal{N}(0, I)$. The denoiser, denoted by ϵ_θ, is then trained by minimizing the following loss function:

$$\mathcal{L}_{ldm} = \|\epsilon_\theta(z_t, t, c) - \epsilon\|^2$$

Here, c is the conditioning vector. In the case of a text-to-image diffusion model, c is the output of a text encoder that predicts the textual embedding of the image caption.

3.2.2. Using real Data as Anchors

Several works in the literature have investigated the potential of synthetic images for training downstream classifiers [8, 12, 30]. A consistent finding across these studies is that synthetic images can improve classifier performance when augmented with real data. However, even state-of-the-art synthetic images exhibit a slight distribution shift when not anchored to real images [8, 22]. Consequently, real data and synthetic samples augmented from real data tend to outperform purely synthetic samples.

In this work, we propose to use the low-resolution patches stored during the coreset selection step as anchors on the manifold of training distribution. Let $x_i^* \in \mathbb{R}^{\lfloor H/r \rfloor \times \lfloor W/r \rfloor \times 3}$ be the ith low-resolution patch subsampled in the previous step. We propose to first upsample the patch using 2D interpolation to the original image size:

$$\hat{x}_i^* = \text{INTERP}(x_i^*),$$

where $\hat{x}_i^* \in \mathbb{R}^{H \times W \times 3}$. The upsampled patch matches the original image dimensions in $\mathcal{D}$ but still retains low quality. We propose using this low-quality image as an anchor for the LDM. Let $\hat{z}_i^*$ be the latent code of the upsampled patch. We first add a small amount of noise to $\hat{z}_i^*$:

$$\hat{z}_{i,t'}^* = \sqrt{\bar{\alpha}_{t'}} \hat{z}_i^* + \sqrt{1 - \bar{\alpha}_{t'}}\epsilon, \tag{2}$$

where $t' \in \{1, 2, \ldots, T\}$. We define ρ as $\frac{t'}{T} \in [0, 1]$ and use it as a hyperparameter that controls the amount of noise. This

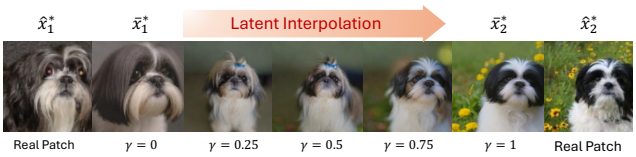

Figure 3. Performing mixup in the latent space of the LDM's autoencoder.

small noise addition perturbs the data slightly off the training data distribution manifold. We then iteratively denoise the low-quality latent back to the manifold using ϵ_θ, following the backward diffusion process:

$$\hat{z}^*_{i,t'-1} = \frac{1}{\sqrt{\alpha_{t'}}}\left(\hat{z}^*_{i,t'} - \frac{1-\alpha_{t'}}{\sqrt{1-\bar{\alpha}_{t'}}}\epsilon_\theta(\hat{z}^*_{i,t'}, t', c)\right) + \sigma_t \epsilon'$$

where $\epsilon' \sim \mathcal{N}(0, I)$, and $\bar{\alpha}_{t'} = \prod_{s=1}^{t'} \alpha_s$. The denoised sample is denoted as $\bar{z}^*_i = \hat{z}^*_{i,0}$.

The denoised image $\bar{x}^*_i = \text{Decode}(\bar{z}^*_i)$ possesses two key properties: 1) since the denoiser is trained on high-resolution images, the denoised image will also be high-resolution, making $\bar{x}^*_i$ high quality; and 2) as a projection onto the manifold of the training distribution, $\bar{x}^*_i$ does not necessarily recover the same anchor patch, with the contents of $\bar{x}^*_i$ varying slightly based on c and ρ. Therefore, the final transformation results in a combination of super-resolution and semantic augmentation.

3.3. Mixup in Latent Space

Mixup [38] has become a widely adopted data augmentation method for training vision models. It encourages local linearity in the model by enforcing that the linear mixture of input images corresponds to the linear mixture of their outputs. This concept was later extended to manifold mixup [32], which better aligns with the underlying data manifold. In our approach, we assume that both the student and teacher models have access to an expressive LDM. This allows us to leverage the LDM to perform mixup operations in the latent space, effectively implementing manifold mixup, to further augment the student's limited set of samples. Let $\hat{z}^*_1$ represent the latent code of an upsampled patch in the training data belonging to class k. To augment this patch, we randomly sample another data point $\hat{z}^*_2$ from the same class and perform linear interpolation in the latent space with a mixing parameter γ, defined as $\hat{z}_{\text{interp}} = \gamma \hat{z}^*_1 + (1-\gamma)\hat{z}^*_2$. The interpolated latent code is then used for augmentation via the LDM. Figure 3 presents qualitative results of the augmented samples generated using mixup. In our ablation study, we highlight the performance improvements achieved by mixing latent codes.

3.4. Putting it all together

For each IPC, we extract r^2 low-resolution patches from our coreset selection. For each patch, we generate m high-quality augmentations using the diffusion model and pass them through the teacher model to obtain soft labels, resulting in a total of $m \times r^2$ soft labels. In all our experiments, we set m equal to the number of training epochs for the student. Notably, regenerating the high-quality augmentations requires only a single random seed. Lastly, we emphasize that the storage overhead for soft labels is present in many of the recent methods that combine knowledge distillation and dataset distillation, such as RDED [28], SRe2L [36], and G-VBSM [24].

4. Experiments

We evaluated our method against both knowledge-distillation-based and bilevel-optimization-based approaches across several high-resolution benchmarks:

1. **Tiny-ImageNet [18]:** This dataset includes 200 classes of 64×64 images derived from the original ImageNet-1k dataset. For our experiments, we selected patches at a resolution of 32×32.
2. **ImageWoof and ImageNette [13]:** These datasets are 10-class subsets of ImageNet-1k, with an original resolution of 224×224. ImageWoof focuses on different dog breeds, while ImageNette covers a broad array of categories spanning animals and objects. To ensure comparability with RDED, we used a patch size of 112×112.
3. **ImageNet-1k [5]:** A comprehensive dataset containing 1000 classes of 224×224 images representing a diverse set of categories. Consistent with [28], we employed patches of 112×112.

While our method demonstrates its core strength on high-resolution benchmarks that benefit from super-resolution capabilities, we also benchmarked on CIFAR-10 and CIFAR-100 [16] to address the challenges bilevel methods face with high-resolution images and widely used ResNet architectures. For these datasets, we used 16×16 patches.

The following baseline methods, including both bilevel-optimization-based and knowledge-distillation-based approaches, were used for evaluation:

1. **RDED [28]:** The first method that synthesizes collages of important patches, selected based on the teacher model's cross-entropy loss.
2. **SRe2L [36]:** Leverages batch norm statistics of the teacher model to perform model inversion, facilitating the synthesis of diverse samples.
3. **G-VBSM [24]:** Extending [36], this approach leverages rich statistical information from batch norm layers of multiple pretrained teachers to synthesize data.
4. **MTT [1]:** The first approach to define the objective of outer-level optimization by matching the training trajectory of the student model to the expert's.
5. **IDM [42]:** Proposes efficient data distillation through distribution matching between synthetic and real data.
6. **Tesla [4]:** Simplifies gradient calculations for trajectory-

		MTT	SRe2L	G-VBSM	RDED	Ours
		ConvNet	ResNet-18	ResNet-18	ResNet-18	ResNet-18
Tiny-ImageNet	IPC=1	8.8 ± 0.3	2.62 ± 0.1	—	9.7 ± 0.4	19.4 ± 1.41
	IPC=10	23.2 ± 0.2	16.1 ± 0.2	—	41.9 ± 0.2	46.2 ± 0.06
	IPC=50	28.0 ± 0.3	41.1 ± 0.4	47.6 ± 0.3	58.2 ± 0.1	53.4 ± 0.64
ImageWoof	IPC=1	28.6 ± 0.8	13.3 ± 0.5	—	17.9 ± 1.0	25.0 ± 0.81
	IPC=10	35.8 ± 1.8	20.2 ± 0.2	—	44.4 ± 1.8	47.6 ± 1.27
	IPC=50	—	23.3 ± 0.3	—	71.7 ± 0.3	77.1 ± 0.02
ImageNette	IPC=1	47.7 ± 0.9	19.1 ± 1.1	—	35.8 ± 1.0	51.4 ± 1.49
	IPC=10	63.0 ± 1.3	29.4 ± 3.0	—	61.4 ± 0.4	73.6 ± 0.12
	IPC=50	—	40.9 ± 0.3	—	80.4 ± 0.4	87.6 ± 1.82
ImageNet-1k	IPC=1	—	0.1 ± 0.1	—	6.6 ± 0.2	13.9 ± 0.28
	IPC=10	—	21.3 ± 0.6	31.4 ± 0.5	42.0 ± 0.1	52.1 ± 0.92
	IPC=50	—	46.8 ± 0.2	51.8 ± 0.4	56.5 ± 0.1	54.9 ± 0.45

Table 1. Results on higher-resolution benchmarks. We compared our method against three knowledge-distillation-based approaches and MTT as a bilevel optimization method. Blank cells for the MTT method indicate its lack of scalability, while for G-VBSM represent results not reported by the authors. The results demonstrate that our approach is either superior or on par with the baselines. In all experiments, the teacher and student models share the same architecture. Cross-architectural analysis results are detailed in the ablation studies.

		Tesla	MTT	IDM	DATM	G-VBSM	RDED	Ours	SRe2L	G-VBSM	RDED	Ours
		ConvNet	ConvNet	ConvNet	ConvNet	ConvNet	ConvNet	ConvNet	ResNet18	ResNet18	ResNet18	ResNet18
CIFAR10	IPC=1	48.5	46.3	45.6	46.9	—	23.5	38.2	16.6	—	22.9	31.0
	IPC=10	66.4	65.3	58.6	66.8	46.5	50.2	64.6	29.3	53.5	37.1	47.7
	IPC=50	72.6	71.6	67.5	76.1	54.3	68.4	73.9	45.0	59.2	62.1	70.4
CIFAR100	IPC=1	24.8	24.3	20.1	27.9	16.4	19.6	42.6	6.6	25.9	11.0	31.2
	IPC=10	41.7	40.1	45.1	47.2	38.7	48.1	49.0	31.6	59.5	42.6	57.7
	IPC=50	47.9	47.7	50.0	55.0	45.7	57.0	53.5	50.2	65.0	62.6	62.2

Table 2. Results on CIFAR-10 and CIFAR-100 datasets. We evaluated our method against various bi-level optimization and knowledge-distillation-based approaches using ResNet-18 and ConvNet-3 as the student/teacher architectures. For these low-resolution benchmarks, we stored and communicated 16×16 patches prior to super-resolution and augmentation. Our method matches or outperforms knowledge-distillation approaches. In all experiments, the student and teacher architectures were identical.

matching-based methods, enhancing the computational efficiency of [1].

7. **DATM [9]:** This work improves the trajectory matching methods and aligns the complexity of generated patterns to the dataset's size.

The results are summarized in Table 1 and Table 2. Our approach consistently outperforms knowledge-distillation-based dataset distillation methods (RDED [28], G-VBSM [24], SRe2L [36]) across various large-scale benchmarks under IPC=1 and IPC=10 settings. For IPC=50, our performance matches or exceeds RDED, but the gap narrows for ImageNet-1k, Tiny-ImageNet, and CIFAR-100. This is due to the larger sample count at higher IPCs, especially in datasets with many classes (e.g., 50,000 samples for IPC=50 in ImageNet-1k). As distilled samples increase, LDM augmentation has less impact.

Implementation Details: We conducted all experiments using the float16 variant of the SDXL-Turbo diffusion model, while setting num_inference_steps=5. For most experiments, we employed the AdamW optimizer with a learning rate of 0.001 and a weight decay of 0.01, training for 300 epochs. Empirically, we observed that defining a Bernoulli random variable to determine whether to drop the diffusion augmentation with probability p enhances performance across benchmarks. Through grid search, we found that setting $p = 0.3$, meaning augmentation is dropped for 30% of the batches, yields good results. A more detailed analysis is provided in the ablation studies. Details of each experimental setup and the corresponding hyperparameters are provided in Appendix A1.

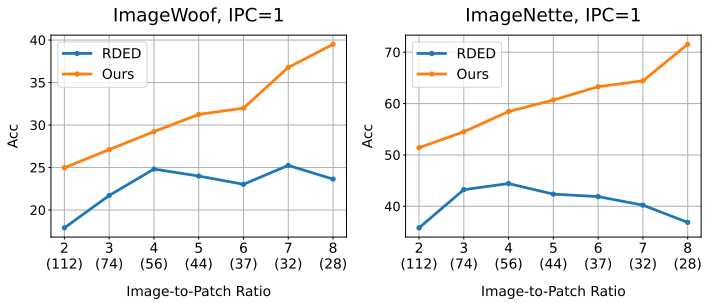

Figure 4. Impact of patch size on student performance in RDED, balancing realism and diversity. The x-axis shows r values, with parentheses indicating patch sizes ($224/r$).

Method	sz112	sz74	sz56	sz44	sz32	sz28
Ours (Acc %)	25.0	27.1	29.2	31.3	36.8	39.5
Ours (Time s)	2.5	2.8	6.4	8.2	16.6	22.5
RDED (Acc %)	17.9	21.7	24.8	24.0	25.3	23.6
RDED (Time s)	1.4	1.5	1.8	2.1	2.9	3.4

Table 3. Comparison of accuracy (%) and training time per epoch (seconds) for different patch sizes across methods.

4.1. Effect of Patch Size

RDED [28] proposes compressing visual information by utilizing patches smaller than the original image dimensions. To explore the performance dynamics associated with varying patch sizes under a fixed memory budget, we conducted a study where, limited to storing one 224×224 image per class, we evaluated performance as patch sizes decreased and their number increased. In the $IPC = 1$ scenario, we stored r^2 patches of size $\frac{H}{r} \times \frac{W}{r}$, with r ranging from 2 to 8. For this analysis, we selected ImageWoof and ImageNette as benchmarks due to their differing classification dynamics: ImageWoof consists of 10 dog breeds with high inter-class similarity, while ImageNette has 10 widely diverse categories with minimal inter-class relationship.

Figure 4 illustrates that in RDED, increasing the number of patches enhances diversity but reduces realism, revealing an optimal performance point at $r = 4$. The reason is that although reducing the patch sizes allows more patches to fit within a fixed memory budget, the patch resolution degrades significantly beyond $r = 4$, leading to a decline in realism, which ultimately harms performance. In contrast, our method demonstrates a different trend: we consistently outperform RDED at all values of r, and our performance continues to improve as the patch count increases. This is achieved through our use of a diffusion model for super-resolution, which not only preserves realism but also enhances diversity by introducing variations to the data, complementing the diversity gains from the increased patch count. However, this improved performance comes at the cost of additional diffusion calls, leading to increased training time. Table 3 presents the average elapsed time per training epoch across various patch sizes.

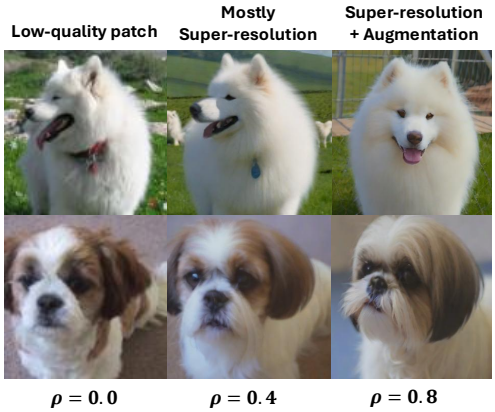

Figure 5. Qualitative illustration of the generated samples. By adjusting ρ, we can control the level of augmentation, allowing us to effectively distinguish between the contributions of super-resolution and augmentation.

4.2. Effect of Super-resolution and Augmentation

Denoising a patch's latent code after partial noise addition enables both content variation and super-resolution. To isolate their effects, we set $\rho = 0.4$ for super-resolution with minimal augmentation and $\rho = 0.8$ for added variation. Figure 5 shows qualitative results.

In Table 4, we illustrate the contribution of various components to overall performance. In "RDED," training is conducted exclusively on real patches. "Only text cond." refers to training on synthetic data using only the textual prompt, "A photo of *category_name*," without storing real patches. Setting $\rho = 0.4$ generates high-quality samples with minimal augmentation, while $\rho = 0.8$ enables both augmentation and super-resolution. The best performance is achieved when super-resolution, augmentation, and latent mixup are incorporated during training.

Components	ImageWoof	ImageNette
RDED	17.9	35.8
Only text cond.	19.6	35.1
Superres ($\rho = 0.4$)	19.5	39.5
Superres+Aug ($\rho = 0.8$)	21.6	47.7
Superres+Aug+Mixup ($\rho = 0.8$)	25.0	51.4

Table 4. Ablation study highlighting the contributions of individual components in our framework and separating the effects of super-resolution from augmentation.

	p=0.	p=0.3	p=0.6	p=0.9	p=1.0 (RDED)
Acc	23.2	25.0	24.3	20.0	17.9

Table 5. Student performance with varying augmentation drop probabilities. The results demonstrate an optimal balance between enhanced augmented samples and real patches, highlighting a sweet spot for peak performance.

	Real Patches	rnd_res_crop h_flip	Diffusion
Accuracy	94.0	71.2	89.86
MPD	3.7	3.4	4.24

Table 6. Impact of different augmentation strategies on diversity and realism. Our method enhances both the diversity and fidelity of real patches compared to traditional geometric augmentations. We generated 10 augmentations per patch using the classic and LDM-based strategies.

4.3. Effect of Augmentation Drop Probability

We define an *augmentation drop* probability, controlling whether each training iteration uses real upsampled patches (RDED) or diffusion-based augmentations. Our grid search found aug_drop=0.3 optimal for most experiments, highlighting the need for a balanced mix of real and synthetic data. We believe that without applying a drop probability, diffusion-based augmentations risk introducing a distribution shift in the training data. However, by retaining the original patches in the training loop, we enhance diversity within the original distribution while preserving fidelity. Importantly, the synthetic augmentations are generated on-the-fly and immediately discarded after gradient computation, leveraging the inference efficiency of few-step diffusion models. Additionally, the total number of seen samples remains consistent with RDED, as we fixed the number of epochs at 300, aligning with prior baselines. We report grid search results for ImageWoof with IPC=1, where aug_drop controls the ratio of generated samples.

4.4. Quantitative Analysis of Diversity and Realism

To evaluate the diffusion model's ability to generate diverse images, we conducted an ImageWoof IPC=10 experiment. After constructing our coreset, we applied three augmentation strategies: 1) no augmentations, 2) random_resize_crop + horizontal_flipping, and 3) LDM-based augmentation. Diversity was quantified using mean pairwise distance (MPD) in the embedding space of a ResNet-18 pretrained on ImageNet-1k, while fidelity to the original distribution was evaluated through teacher accuracy. The results presented in Table 6 show that, compared to real patches without any augmentations, diffusion-based augmentations enhance diversity with minimal impact on teacher model accuracy. Additionally, compared to the standard rnd_res_crop + h_flip augmentations, LDM improves diversity while maintaining greater fidelity to the natural patch distribution.

4.5. Cross-architectural Analysis

Bilevel-optimization-based methods often suffer poor cross-architectural transferability: student accuracy drops sharply when its architecture differs from the expert's. GLaD [2] mitigates this via a generative prior that enhances realism. Since our method also relies on a generative prior, we compare against GLaD in both high- and low-resolution settings.

	ImageNette	ImageWoof
MTT [1]	24.1	16.0
MTT + **GLaD**	30.4	17.1
DC [41]	28.2	17.4
DC + **GLaD**	31.0	17.8
DM [40]	20.6	14.5
DM + **GLaD**	21.9	15.2
Ours	**30.79**	**19.74**

Table 7. Cross-architectural analysis at a resolution of 128×128 with IPC=1. The teacher model is a ConvNet, while the student architectures include VGG11, ViT, ResNet18, and AlexNet. The reported results represent the average performance across these architectures. 64×64 patches were communicated to the student models within the memory constraints.

	AlexNet	ResNet18	ViT	Average
MTT	26.8	23.4	21.2	23.8
MTT + **GLaD**	27.9	30.2	22.7	26.9
DC	25.9	27.3	22.9	25.4
DC + **GLaD**	26.0	27.6	**23.4**	25.7
DM	22.9	22.2	21.3	22.1
DM + **GLaD**	25.1	22.5	23.0	23.5
Ours	**31.3**	**36.0**	21.9	**29.7**

Table 8. Cross-architectural analysis on the CIFAR-10 dataset using a ConvNet as the teacher and various student architectures. The results demonstrate superior overall cross-architecture performance achieved by our method.

Table 7 reports IPC=1 results with a ConvNet teacher, averaged over four student architectures: VGG11 [26], ViT [6], AlexNet [17], and ResNet-18 [10]. As GLaD does not scale to 224×224, ImageNette and ImageWoof are downsampled to 128×128 for a fair comparison. Table 8 reports the analogous results on CIFAR-10. Both confirm superior cross-architecture performance of our method.

5. Conclusion

Recent dataset distillation work highlights the importance of both realism and diversity in distilled data. Building on these insights, we leverage modern Latent Diffusion Models (LDMs) to address both objectives: combining coreset selection with on-the-fly generative super-resolution, augmentation, and latent-space mixup. The result is state-of-the-art performance across several large-scale dataset distillation benchmarks, demonstrating that LDMs can substantially improve dataset compression while preserving the essential attributes of the data.

References

[1] George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A Efros, and Jun-Yan Zhu. Dataset distillation by matching training trajectories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4750–4759, 2022. 2, 5, 6, 8

[2] George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A Efros, and Jun-Yan Zhu. Generalizing dataset distillation via deep generative prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3739–3748, 2023. 2, 8

[3] Xuxi Chen, Yu Yang, Zhangyang Wang, and Baharan Mirzasoleiman. Data distillation can be like vodka: Distilling more times for better quality. In *The Twelfth International Conference on Learning Representations*, 2024. 2

[4] Justin Cui, Ruochen Wang, Si Si, and Cho-Jui Hsieh. Scaling up dataset distillation to imagenet-1k with constant memory. In *International Conference on Machine Learning*, pages 6565–6590. PMLR, 2023. 5

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5

[6] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 8

[7] Jiawei Du, Yidi Jiang, Vincent YF Tan, Joey Tianyi Zhou, and Haizhou Li. Minimizing the accumulated trajectory error to improve dataset distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3749–3758, 2023. 2

[8] Lijie Fan, Kaifeng Chen, Dilip Krishnan, Dina Katabi, Phillip Isola, and Yonglong Tian. Scaling laws of synthetic images for model training... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7382–7392, 2024. 4

[9] Ziyao Guo, Kai Wang, George Cazenavette, Hui Li, Kaipeng Zhang, and Yang You. Towards lossless dataset distillation via difficulty-aligned trajectory matching. *arXiv preprint arXiv:2310.05773*, 2023. 6

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 8

[11] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 2

[12] Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition? *arXiv preprint arXiv:2210.07574*, 2022. 4

[13] Jeremy Howard and Sylvain Gugger. Imagenette: A smaller subset of 10 easily classified classes from imagenet. https://github.com/fastai/imagenette, 2019. Accessed: 2024-11-14. 2, 4, 5

[14] Khawar Islam, Muhammad Zaigham Zaheer, Arif Mahmood, and Karthik Nandakumar. Diffusemix: Label-preserving data augmentation with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27621–27630, 2024. 3

[15] Jang-Hyun Kim, Jinuk Kim, Seong Joon Oh, Sangdoo Yun, Hwanjun Song, Joonhyun Jeong, Jung-Woo Ha, and Hyun Oh Song. Dataset condensation via efficient synthetic-data parameterization. In *International Conference on Machine Learning*, pages 11102–11118. PMLR, 2022. 2

[16] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 and cifar-100 datasets. *URl: https://www. cs. toronto. edu/kriz/cifar. html*, 6(1):1, 2009. 5

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012. 8

[18] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015. 5

[19] Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. *Advances in neural information processing systems*, 34:20596–20607, 2021. 2

[20] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 2

[21] Mattia Sangermano, Antonio Carta, Andrea Cossu, and Davide Bacciu. Sample condensation in online continual learning. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 01–08. IEEE, 2022. 1

[22] Mert Bülent Sarıyıldız, Karteek Alahari, Diane Larlus, and Yannis Kalantidis. Fake it till you make it: Learning transferable representations from synthetic imagenet clones. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8011–8021, 2023. 3, 4

[23] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pages 87–103. Springer, 2025. 2, 3, 4

[24] Shitong Shao, Zeyuan Yin, Muxin Zhou, Xindong Zhang, and Zhiqiang Shen. Generalized large-scale data condensation via various backbone and statistical matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16709–16718, 2024. 2, 3, 5, 6

[25] Jordan Shipard, Arnold Wiliem, Kien Nguyen Thanh, Wei Xiang, and Clinton Fookes. Diversity is definitely needed: Improving model-agnostic zero-shot classification via stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 769–778, 2023. 3

[26] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 8

[27] Felipe Petroski Such, Aditya Rawal, Joel Lehman, Kenneth Stanley, and Jeffrey Clune. Generative teaching networks:

Accelerating neural architecture search by learning to generate synthetic training data. In *International Conference on Machine Learning*, pages 9206–9216. PMLR, 2020. 1

[28] Peng Sun, Bei Shi, Daiwei Yu, and Tao Lin. 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*, pages 9390–9399, 2024. 2, 3, 5, 6, 7

[29] Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J Gordon. An empirical study of example forgetting during deep neural network learning. In *International Conference on Learning Representations*, 2019. 2

[30] Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models. *arXiv preprint arXiv:2302.07944*, 2023. 4

[31] Brandon Trabucco, Kyle Doherty, Max A Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. 3

[32] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International conference on machine learning*, pages 6438–6447. PMLR, 2019. 5

[33] Kai Wang, Bo Zhao, Xiangyu Peng, Zheng Zhu, Shuo Yang, Shuo Wang, Guan Huang, Hakan Bilen, Xinchao Wang, and Yang You. Cafe: Learning to condense dataset by aligning features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12196–12205, 2022. 2

[34] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A Efros. Dataset distillation. *arXiv preprint arXiv:1811.10959*, 2018. 1, 2

[35] Yuanhao Xiong, Ruochen Wang, Minhao Cheng, Felix Yu, and Cho-Jui Hsieh. Feddm: Iterative distribution matching for communication-efficient federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16323–16332, 2023. 1

[36] Zeyuan Yin, Eric Xing, and Zhiqiang Shen. Squeeze, recover and relabel: Dataset condensation at imagenet scale from a new perspective. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3, 5, 6

[37] Ruonan Yu, Songhua Liu, and Xinchao Wang. Dataset distillation: A comprehensive review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1

[38] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. 5

[39] Yifan Zhang, Daquan Zhou, Bryan Hooi, Kai Wang, and Jiashi Feng. Expanding small-scale datasets with guided imagination. *Advances in neural information processing systems*, 36:76558–76618, 2023. 3

[40] Bo Zhao and Hakan Bilen. Dataset condensation with distribution matching. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 6514–6523, 2023. 2, 8

[41] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. Dataset condensation with gradient matching. In *Ninth International Conference on Learning Representations 2021*, 2021. 2, 8

[42] Ganlong Zhao, Guanbin Li, Yipeng Qin, and Yizhou Yu. Improved distribution matching for dataset condensation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7856–7865, 2023. 5

[43] Daquan Zhou, Kai Wang, Jianyang Gu, Xiangyu Peng, Dongze Lian, Yifan Zhang, Yang You, and Jiashi Feng. Dataset quantization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17205–17216, 2023. 2

Appendix

A1. Hyperparameters

	Dataset	Patch size	Noise Strength	Temp	aug_drop
CIFAR10	IPC=1	16	1.0	20	0.3
	IPC=10	16	1.0	20	0.3
	IPC=50	16	1.0	20	0.3
CIFAR100	IPC=1	16	1.0	20	0.3
	IPC=10	16	1.0	20	0.75
	IPC=50	32 (Conv) 16 (Res)	1.0	20	0.3
Tiny-ImageNet	IPC=1	32	1.0	20	0.3
	IPC=10	32	1.0	20	0.3
	IPC=50	32	1.0	20	0.3
ImageNet-1k	IPC=1	112	0.8	20	0.3
	IPC=10	112	0.8	20	0.3
	IPC=50	112	0.8	20	0.5
ImageWoof	IPC=1	112	0.8	5	0.3
	IPC=10	112	0.8	20	0.3
	IPC=50	112	0.8	20	0.3
ImageNette	IPC=1	112	0.8	5	0.3
	IPC=10	112	0.8	20	0.3
	IPC=50	112	0.8	20	0.3

Table 9. Hyperparameter configurations for different datasets with varying IPC values.

Table 9 lists the detailed hyperparameters used in our experiments. For all experiments, we used the AdamW optimizer with a learning rate of 0.001 and a weight decay of 0.01.

A2. Different Domains

We also investigated the use of the pretrained LDM for augmenting images that slightly deviate from the natural image distribution. To assess our approach, we evaluated it on the UCMerced Land Use dataset and compared against RDED. The dataset comprises 21 classes of aerial images, with 100 images per class at a resolution of 256×256. The results, presented in Table 10, include evaluations under IPC=1 and IPC=10 settings. The 21 class names are: agricultural, airplane, baseball-diamond, beach, buildings, chaparral, denseresidential, forest, freeway, golfcourse, harbor, intersection, mediumresidential, mobilehomepark, overpass, parkinglot, river, runway, sparseresidential, storagetanks, tenniscourt.

Method	IPC=1	IPC=10
RDED	26.6	63.1
Ours	**35.3**	**64.2**

Table 10. Comparison against RDED on the UCMerced Land Use dataset. Patch sizes are 128×128.

A3. One-hot vs Soft-Labels and Patch Selection Strategy

Here, we highlight the performance difference between using teacher soft labels and one-hot encoded labels. Additionally, we conducted an ablation study to assess the impact of randomly selecting patches from the dataset versus filtering them based on the cross-entropy criterion described in the paper. The experiment was conducted on ImageWoof with IPC=10, comparing

three baselines: randomly selected patches, CE-selected patches (RDED), and CE-selected patches with on-the-fly diffusion augmentations (Ours). The results are presented in Table 11.

Acc (std)	Random imgs	RDED	Ours
one-hot	20.2 (0.6)	28.7 (1.3)	**37.8 (0.2)**
soft-label	25.5 (1.0)	44.1 (1.8)	**47.6 (1.1)**

Table 11. Ablation over different patch selection strategies and label types.

Table 11 shows that our method outperforms RDED by a larger margin when using one-hot encoded labels. We believe this is because one-hot labels carry significantly less information than teacher soft labels, making the student model more dependent on the quality of the distilled samples. Within our framework, as demonstrated in the paper, both the fidelity and diversity of the synthetically augmented patches are superior to the traditional geometric augmentations used in RDED.