Diversity-Driven Synthesis: Enhancing Dataset Distillation through Directed Weight Adjustment

Jiawei Du[1,2] **Xin Zhang**[1,2,3] **Juncheng Hu**[4] **Wenxing Huang**[1,2,5] **Joey Tianyi Zhou**[1,2]✉
[1] Centre for Frontier AI Research (CFAR), Agency for Science, Technology and Research (A*STAR), Singapore
[2] Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), Singapore
[3]XiDian University, Xi'an, China [4]National University of Singapore, Singapore
[5]Hubei University, WuHan, China

Abstract

The sharp increase in data-related expenses has motivated research into condensing datasets while retaining the most informative features. Dataset distillation has thus recently come to the fore. This paradigm generates synthetic datasets that are representative enough to replace the original dataset in training a neural network. To avoid redundancy in these synthetic datasets, it is crucial that each element contains unique features and remains diverse from others during the synthesis stage. In this paper, we provide a thorough theoretical and empirical analysis of diversity within synthesized datasets. We argue that enhancing diversity can improve the parallelizable yet isolated synthesizing approach. Specifically, we introduce a novel method that employs dynamic and directed weight adjustment techniques to modulate the synthesis process, thereby maximizing the representativeness and diversity of each synthetic instance. Our method ensures that each batch of synthetic data mirrors the characteristics of a large, varying subset of the original dataset. Extensive experiments across multiple datasets, including CIFAR, Tiny-ImageNet, and ImageNet-1K, demonstrate the superior performance of our method, highlighting its effectiveness in producing diverse and representative synthetic datasets with minimal computational expense. Our code is available at https://github.com/AngusDujw/Diversity-Driven-Synthesis.

1 Introduction

With the rapid growth in dataset size and the need for efficient data storage and processing [8, 17, 14, 13], how to condense datasets while preserving their key characteristics becomes a significant challenge in machine learning community [12, 38]. Unlike previous research [29, 39, 50, 44] that focuses on constructing a representative subset through selecting from the original data, *Dataset Distillation* [43, 31, 20] aims to synthesize a small and compact dataset that retains informative features from the original dataset. A model trained on the synthetic dataset is thus supposed to achieve comparable performance as one trained on the original dataset. The development of dataset distillation reduces data-related costs [7, 34, 49] and helps us better understand how Deep Neural Networks (DNNs) extract knowledge from large-scale datasets.

Numerous studies dedicate significant effort to synthesizing distilled datasets more effectively. For example, Zhao *et al.* employ a gradient-matching approach [52, 54] to guide the synthesis process. Trajectory-matching methods [1, 2, 5, 6] further align gradient trajectories to optimize the synthetic data. Additionally, distribution matching [42, 53, 55] and kernel inducing points methods [28, 25, 23, 24] also contribute to synthesizing representative data. Despite the great progress achieved by these methods on datasets like CIFAR [16], their extensive computational overhead (both GPU memory and GPU time) hinders the extension of these methods to large-scale datasets like ImageNet-1K [3].

Email: dujiawei@u.nus.edu, joey.tianyi.zhou@gmail.com. ✉ represents the corresponding author.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

https://doi.org/10.52202/079017-3796

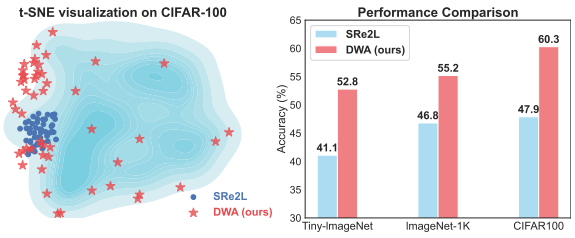

Figure 1: **Left:** t-SNE visualization of logit embeddings on CIFAR-100 [16] dataset. The scatter plot illustrates the distribution of synthetic data instances distilled by SRe2L (blue dots) and our DWA method (red stars). The blue density contours represent the distribution of natural data instances. Our DWA method demonstrates a more diverse and widespread distribution compared to SRe2L [46], indicating better generalization and coverage of the feature space. **Right:** The consequent performance improvement of DWA in various datasets. Experiments are conducted with 50 images per class.

Several recent works [2, 46, 22, 51, 57] have attempted to address the efficiency issues of dataset distillation. In particular, Yin *et al.* [46] propose a lightweight distillation method, SRe2L, which successfully condenses the large-scale dataset ImageNet-1K. Unlike previous methods [1, 53, 15] that treat the synthetic set as a unified entity to utilize the mutual influences among synthetic instances, SRe2L synthesizes each synthetic data instance individually. As such, SRe2L significantly reduces both GPU memory costs and computational overhead.

Individually synthesizing each data instance can efficiently parallelize optimization tasks, thereby flexibly managing GPU memory usage and computational overhead. However, this approach may present challenges in ensuring the representativeness and diversity of each instance. If each instance is synthesized in isolation, there may be a risk of missing the holistic view of the data characteristics, which is crucial for the training of generalized neural networks. Intuitively, SRe2L might expect that random initialization of synthetic data would provide sufficient diversity to prevent homogeneity in the synthetic dataset. Nevertheless, our analysis, as demonstrated in Figure 1, reveals that this initialization contributes only marginally to diversity. Conversely, the Batch Normalization (BN) loss [45] in SRe2L plays the practical role in enhancing diversity of the distilled dataset.

Motivated by these findings, we further investigate the factors that enhance the diversity of synthetic datasets from a theoretical perspective. We reveal that the variance regularizer in the BN loss is the key factor ensuring diversity. Conversely, the mean regularizer within the same BN loss unexpectedly constrains diversity. To resolve this contradiction, we suggest a decoupled coefficient to specifically strengthen the variance regularizer's role in promoting diversity. Experimental results validate our hypothesis. We further propose a dynamic mechanism to adjust the weight parameters of the teacher model. Serving as the sole source of supervision from the original dataset, the teacher model guides the synthesis comprehensively. Our meticulously designed weight perturbation mechanism injects randomness without compromising the informative supervision, thereby improving overall performance. Importantly, our method incurs negligible additional computations ($< 0.1\%$). Intuitively, our method perturbs the weight in a direction that reflects the characteristics of a large subset, varying with each batch of synthesized data.

We conduct extensive experiments across various datasets, including CIFAR-10, CIFAR-100, Tiny-ImageNet, and ImageNet-1K, to verify the effectiveness of our proposed method. The superior performance of our method not only validates our hypothesis but also demonstrates its ability to enhance the diversity of synthetic datasets. This success guides further investigations into searching for representative synthetic datasets for lossless dataset distillation. Our contribution can be summarized as follows:

- We analyze the diversity of the synthetic dataset in dataset distillation both theoretically and empirically, identifying the importance of ensuring diversity in isolated synthesizing approaches.
- We propose a dynamic adjustment mechanism to enhance the diversity of the synthesized dataset, incurring negligible additional computations while significantly improving overall performance. Extensive experiments on various datasets verify the remarkable performance of our method.

2 Preliminaries

Notation and Objective. Given a real and large dataset $\mathcal{T} = \{(\tilde{\boldsymbol{x}}_i, \boldsymbol{y}_i)\}_{i=1}^{|\mathcal{T}|}$, Dataset Distillation aims to synthesize a tiny and compact dataset $\mathcal{S} = \{(\tilde{\boldsymbol{s}}_i, \boldsymbol{y}_i)\}_{i=1}^{|\mathcal{S}|}$. The samples in $\mathcal{T}$ are drawn i.i.d from a natural distribution $\mathcal{D}$, while the samples in $\mathcal{S}$ are optimized from scratch. We use $\theta_{\mathcal{T}}$ and

https://doi.org/10.52202/079017-3796

$\theta_{\mathcal{S}}$ to represent the converged weight trained on $\mathcal{T}$ and $\mathcal{S}$, respectively. We define a neural network $h = g \circ f$, where g acts as the feature extractor and f as the classifier. The feature extractor and the classifier loaded with the corresponding weight parameters from θ are denoted by g_θ and f_θ.

Throughout the paper, we explore the properties of synthesized datasets within the latent space. We transform both $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{s}} \in \mathbb{R}^{\mathrm{C}\times\mathrm{H}\times\mathrm{W}}$ from the pixel space, to the latent space, $\boldsymbol{x}, \boldsymbol{s} \in \mathbb{R}^d$, for better formulation. This transformation is given by $\boldsymbol{x} = g_{\theta_\mathcal{T}}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{s} = g_{\theta_\mathcal{T}}(\tilde{\boldsymbol{s}})$. The objective of Dataset Distillation is to ensure that a model h trained on the synthetic dataset $\mathcal{S}$ is able to achieve a comparable test performance as the model trained with $\mathcal{T}$, which can be formulated as,

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}}\left[\ell\left(h_{\theta_\mathcal{T}}, \boldsymbol{x}\right)\right] \simeq \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}}\left[\ell\left(h_{\theta_\mathcal{S}}, \boldsymbol{x}\right)\right], \tag{1}$$

where ℓ can be an arbitrary loss function. The expression $\ell(h_{\theta_\mathcal{T}}, \boldsymbol{x})$ should be interpreted as $\ell(h_{\theta_\mathcal{T}}, \boldsymbol{x}, \boldsymbol{y})$, where $\boldsymbol{y}$ is the ground truth label.

Synthesizing $\mathcal{S}$. A series of previous works mentioned in Section 5 have introduced various methods to synthesize $\mathcal{S}$. Specifically, SRe2L [46] proposes an efficient and effective synthesizing method, which optimizes each synthetic instance $\boldsymbol{s}_i$ by solving the following minimization problem:

$$\mathop{\arg\min}_{\boldsymbol{s}_i\in\mathbb{R}^d}\left[\ell\left(f_{\theta_\mathcal{T}}, \boldsymbol{s}_i\right) + \lambda\mathcal{L}_{\mathrm{BN}}\left(f_{\theta_\mathcal{T}}, \boldsymbol{s}_i\right)\right], \tag{2}$$

where $\mathcal{L}_{\mathrm{BN}}$ denotes the BN loss, and λ is the coefficient of $\mathcal{L}_{\mathrm{BN}}$. The detailed definition of $\mathcal{L}_{\mathrm{BN}}$ can be found in Equation 3. Minimizing the BN loss $\mathcal{L}_{\mathrm{BN}}$ significantly enhances the performance of SRe2L, which is designed to ensure that $\mathcal{S}$ aligns with the same normalization distribution as $\mathcal{T}$. However, we argue that another essential but overlooked aspect of the BN loss $\mathcal{L}_{\mathrm{BN}}$ is its role in introducing diversity to $\mathcal{S}$, which also greatly benefits the final performance. In the following section, we will analyze this issue in greater detail.

3 Methodology

Diversity in the synthetic dataset $\mathcal{S}$ is essential for effective use of the limited distillation budget. This section reveals that the BN loss, referenced in Equation 2, enhances $\mathcal{S}$'s diversity. However, the suboptimal setting of BN loss limits this diversity. To overcome this, we propose a dynamic adjustment mechanism for the weight parameters of $f_{\theta_\mathcal{T}}$, enhancing diversity during synthesis. Finally, we detail our algorithm and theoretically demonstrate its effectiveness. The pseudocode of our proposed DWA can be found in Algorithm 1.

Algorithm 1 Directed Weight Adjustment (DWA)

Input: Original dataset $\mathcal{T}$; Number of iterations T; Image per class `ipc`; Number of steps K, magnitude ρ to solve the weight adjustment $\widetilde{\Delta\theta}$; Learning rate η; A network $f_{\theta_\mathcal{T}}$ with weight parameter $\theta_\mathcal{T}$, $f_{\theta_\mathcal{T}}$ is well trained on $\mathcal{T}$.

Initialize $\mathcal{S} = \{\}$, $\Delta\theta_0 = \mathbf{0}_{\dim(\theta_\mathcal{T})}$
for $i = 1$ to `ipc` **do**
  Randomly select one instance for each class from $\mathcal{T}$, to initialize $\mathcal{S}_0^i$, *i.e.*,
  $\mathcal{S}_0^i = \{(\boldsymbol{x}_i, \boldsymbol{y}_i) \mid (\boldsymbol{x}_i, \boldsymbol{y}_i) \in \mathcal{T}$ and each $\boldsymbol{y}_i$ is unique$\}$
  ▷ Compute the adjustment of weights $\Delta\theta$ by solving Equation 11
  for $k = 1$ to K **do**
    $\Delta\theta_k = \Delta\theta_{k-1} + \frac{\rho}{K}\nabla L_{\mathcal{S}_0^i}(f_{\theta_\mathcal{T}+\Delta\theta_{k-1}})$
  $\widetilde{\Delta\theta} = \Delta\theta_K$ ▷ Directed Weight Adjustment
  ▷ Optimize $\mathcal{S}^i$
  for $t = 1$ to T **do**
    $\mathcal{S}_t^i = \mathcal{S}_{t-1}^i + \eta\nabla_\mathcal{S}\mathcal{L}(f_{\theta_\mathcal{T}+\widetilde{\Delta\theta}}, \mathcal{S}_{t-1}^i)$ ▷ $\mathcal{L}$ is defined in Equation 15
  $\mathcal{S} = \mathcal{S} \cup \{\mathcal{S}^i\}$

Output: Synthetic dataset $\mathcal{S}$

In the actual optimization process, operations occur within the pixel space using the entire network $h_{\theta_\mathcal{T}}$. However, as we discuss the optimization in the latent space, we only consider solutions within this space. Then, we transform the solution in latent space back into pixel space as $\tilde{\boldsymbol{s}} = g_{\theta_\mathcal{T}}^{-1}(\boldsymbol{s})$.

https://doi.org/10.52202/079017-3796

3.1 Batch Normalization Loss Enhances Diversity of $\mathcal{S}$

The BN loss $\mathcal{L}_{\text{BN}}$ comprises mean ($\mathcal{L}_{\text{mean}}$) and variance ($\mathcal{L}_{\text{var}}$) components, defined as follows:

$$\mathcal{L}_{\text{BN}} = \mathcal{L}_{\text{mean}} + \mathcal{L}_{\text{var}} \quad \text{where} \quad \mathcal{L}_{\text{mean}}\left(f_{\theta_{\mathcal{T}}}, \boldsymbol{s}_i\right) = \sum\nolimits_l \left\|\mu_l\left(\mathbb{S}\right) - \mu_l\left(\mathcal{T}\right)\right\|_2, \\ \text{and} \quad \mathcal{L}_{\text{var}}\left(f_{\theta_{\mathcal{T}}}, \boldsymbol{s}_i\right) = \sum\nolimits_l \left\|\sigma_l^2\left(\mathbb{S}\right) - \sigma_l^2\left(\mathcal{T}\right)\right\|_2, \tag{3}$$

where μ_l and σ_l^2 refer to the channel mean and variance in the l-th layer, respectively. $\boldsymbol{s}_i$ is optimized within a mini-batch $\mathbb{S}$, where $\boldsymbol{s}_i \in \mathbb{S}$ and $\mathbb{S} \subset \mathcal{S}$. Each component of $\mathcal{L}_{\text{BN}}$ operates from its own perspective to enhance dataset distillation. First, the mean component $\mathcal{L}_{\text{mean}}$ regularizes the synthetic data $\boldsymbol{s}$, ensuring its values align closely with those of the representative centroid of $\mathcal{T}$ in latent space. Second, the variance component $\mathcal{L}_{\text{var}}$ encourages the synthetic data in $\mathbb{S}$ to differ from each other, thereby maintaining the variance $\sigma_l^2(\mathbb{S})$. Thus, this BN loss-driven synthesis can be decoupled as

$$\boldsymbol{s}_i = \boldsymbol{X}_c\left(\lambda\mathcal{L}_{\text{mean}}, \theta_{\mathcal{T}}\right) + \boldsymbol{\xi}_i, \tag{4}$$

where $\boldsymbol{X}_c$ can be regarded as an optimal solution to Equation 2 when the variance regularization term $\mathcal{L}_{\text{var}}$ is not considered, *i.e.*,

$$\left\|\nabla_\theta \ell\left(f_{\theta_{\mathcal{T}}}, \boldsymbol{X}_c\right)\right\|_2 \leq \alpha_1 \quad \text{and} \quad \mathcal{L}_{\text{mean}}\left(f_{\theta_{\mathcal{T}}}, \boldsymbol{X}_c\right) = \sum\nolimits_l \left\|\mu_l\left(\boldsymbol{X}_c\right) - \mu_l\left(\mathcal{T}\right)\right\|_2 \leq \alpha_2, \tag{5}$$

where both $\alpha_1, \alpha_2 > 0$ and $\alpha_1, \alpha_2 \to 0$. $\boldsymbol{\xi}_i$ represents a small perturbation and $\boldsymbol{\xi}_i \sim \mathcal{N}\left(0, \sigma_{\boldsymbol{\xi}}^2(\lambda\mathcal{L}_{\text{var}})\right)$. Therefore, the variance of the synthetic dataset $\mathcal{S}$ is,

$$\text{Var}(\mathcal{S}) = \text{Var}\big(\boldsymbol{X}_c(\lambda\mathcal{L}_{\text{mean}}, \theta_{\mathcal{T}})\big) + \text{Var}\left(\boldsymbol{\xi}\right) = \sigma_{\boldsymbol{\xi}}^2\left(\lambda\mathcal{L}_{\text{var}}\right). \tag{6}$$

We have $\text{Var}\big(\boldsymbol{X}_c(\lambda\mathcal{L}_{\text{mean}}, \theta_{\mathcal{T}})\big) = 0$ as $\boldsymbol{X}_c$ is deterministic. Unlike other approaches that consider the mutual influences among synthetic data instances and optimize the dataset collectively, SRe2L [46] optimizes each synthetic data instance individually. Therefore, the diversity of the synthetic dataset $\mathcal{S}$ is solely determined by $\lambda\mathcal{L}_{\text{var}}$.

However, simply increasing λ contributes marginally to enhancing the diversity of $\mathcal{S}$. This is because a greater λ will also emphasize the regularization term $\lambda\mathcal{L}_{\text{mean}}$, which contradicts the emphasis on $\lambda\mathcal{L}_{\text{var}}$. We provide a detailed analysis in the Appendix A.1. As a result, we propose using a decoupled coefficient, λ_{var}, to enhance the diversity of $\mathcal{S}$.

Additionally, the synthetic data instances are optimized individually to approximate the representative data instance $\boldsymbol{X}_c$. However, the gaussian initialization $\mathcal{N}(0, 1)$ in pixel space does not distribute uniformly around $\boldsymbol{X}_c$ in latent space, making the converged synthetic data instances to cluster in a crowed area in latent space, as dedicated in Figure 1. To address this, we propose initializing with real instances from $\mathcal{T}$ inspired by MTT [1], ensuring a uniform projection when synthesizing $\mathcal{S}$.

3.2 Random Perturbation on $\theta_{\mathcal{T}}$ Helps Improve Diversity

In the previous section, we highlighted the often overlooked aspect of the BN loss in introducing diversity to $\mathcal{S}$, which was also verified through experiments in Section 4.2. Building upon this, we propose to introduce randomness into $\theta_{\mathcal{T}}$ to further enhance $\mathcal{S}$'s diversity, as it is the only remaining factor affecting $\text{Var}(\mathcal{S})$, as shown in Equation 6.

Let $\boldsymbol{x}_c^* = \boldsymbol{X}_c(\lambda\mathcal{L}_{\text{mean}}, \theta_{\mathcal{T}})$ to be the original optimal solution to Equation 2. We aim to solve the adjusted optimal solution $\boldsymbol{x}_c = \boldsymbol{X}_c(\lambda\mathcal{L}_{\text{mean}}, \theta_{\mathcal{T}} + \Delta\theta) = \boldsymbol{x}_c^* + \Delta\boldsymbol{x}$, where $\theta_{\mathcal{T}}$ is randomly perturbed by $\Delta\theta$, and $\Delta\theta \sim \mathcal{N}(0, \sigma_\theta^2)$. Consequently, we have:

$$\left\|\nabla_\theta \ell\left(f_{\theta_{\mathcal{T}}+\Delta\theta}, \boldsymbol{x}_c\right)\right\|_2 = \left\|\nabla_\theta \ell\left(f_{\theta_{\mathcal{T}}+\Delta\theta}, \boldsymbol{x}_c^* + \Delta\boldsymbol{x}\right)\right\|_2 \leq \alpha_1. \tag{7}$$

To solve for $\Delta\boldsymbol{x}$, we can apply a first-order bivariate Taylor series approximation because $\nabla_\theta \ell(f_{\theta_{\mathcal{T}}}, \boldsymbol{X}_c) \leq \alpha_1$, where $\alpha_1 \to 0$, and both $\Delta\theta$ and $\Delta\boldsymbol{x}$ are small. Thus,

$$\begin{aligned} &\left\|\nabla_\theta \ell\left(f_{\theta_{\mathcal{T}}+\Delta\theta}, \boldsymbol{x}_c^* + \Delta\boldsymbol{x}\right)\right\|_2 \\ =& \left\|\nabla_\theta \ell\left(f_{\theta_{\mathcal{T}}}, \boldsymbol{x}_c^*\right) + \nabla_\theta^2 \ell\left(f_{\theta_{\mathcal{T}}}, \boldsymbol{x}_c^*\right)\Delta\theta + \nabla_{\boldsymbol{x}}\left[\nabla_\theta \ell\left(f_{\theta_{\mathcal{T}}}, \boldsymbol{x}_c^*\right)\right]\Delta\boldsymbol{x}\right\|_2 \\ \leq& \left\|\nabla_\theta \ell\left(f_{\theta_{\mathcal{T}}}, \boldsymbol{x}_c^*\right)\right\|_2 + \left\|\nabla_\theta^2 \ell\left(f_{\theta_{\mathcal{T}}}, \boldsymbol{x}_c^*\right)\Delta\theta + \nabla_{\boldsymbol{x}}\left[\nabla_\theta \ell\left(f_{\theta_{\mathcal{T}}}, \boldsymbol{x}_c^*\right)\right]\Delta\boldsymbol{x}\right\|_2 \\ \leq& \alpha_1 + \left\|\nabla_\theta^2 \ell\left(f_{\theta_{\mathcal{T}}}, \boldsymbol{x}_c^*\right)\Delta\theta + \nabla_{\boldsymbol{x}}\left[\nabla_\theta \ell\left(f_{\theta_{\mathcal{T}}}, \boldsymbol{x}_c^*\right)\right]\Delta\boldsymbol{x}\right\|_2, \end{aligned} \tag{8}$$

We disregard the class differences in the following analysis since they are identical across all classes.

To satisfy Equation 7, we have:

$$\nabla_\theta^2 \ell\left(f_{\theta_\mathcal{T}}, \boldsymbol{x}_c^*\right) \Delta\theta + \nabla_{\boldsymbol{x}}\left[\nabla_\theta \ell\left(f_{\theta_\mathcal{T}}, \boldsymbol{x}_c^*\right)\right] \Delta\boldsymbol{x} = \mathbf{0}, \quad \text{then}$$
$$\Delta\boldsymbol{x} = -\nabla_{\boldsymbol{x}}\left[\nabla_\theta \ell\left(f_{\theta_\mathcal{T}}, \boldsymbol{x}_c^*\right)\right]^{-1} \nabla_\theta^2 \ell\left(f_{\theta_\mathcal{T}}, \boldsymbol{x}_c^*\right) \Delta\theta. \tag{9}$$

Intuitively, $\Delta\boldsymbol{x}$ must compensate for the ∇_θ incurred by introducing the random perturbation $\Delta\theta \sim \mathcal{N}(0, \sigma_\theta^2)$ on $\theta_\mathcal{T}$. By Equation 9, $\mathrm{Var}(\Delta\boldsymbol{x}) \propto \mathrm{Var}(\Delta\theta) = \sigma_\theta^2$, then:

$$\begin{aligned}
\mathrm{Var}\left(\mathcal{S}'\right) &= \mathrm{Var}\big(\boldsymbol{X}_c\left(\lambda\mathcal{L}_{\text{mean}}, \theta_\mathcal{T} + \Delta\theta\right)\big) + \mathrm{Var}\left(\boldsymbol{\xi}\right) \\
&= \mathrm{Var}\left(\boldsymbol{x}_c^* + \Delta\boldsymbol{x}\right) + \mathrm{Var}\left(\boldsymbol{\xi}\right) \\
&= \beta\sigma_\theta^2 + \sigma_{\boldsymbol{\xi}}^2\left(\lambda\mathcal{L}_{\text{var}}\right) \geq \sigma_{\boldsymbol{\xi}}^2\left(\lambda\mathcal{L}_{\text{var}}\right),
\end{aligned} \tag{10}$$

where β is determined by $-\nabla_{\boldsymbol{x}}[\nabla_\theta \ell(f_{\theta_\mathcal{T}}, \boldsymbol{x}^c)]^{-1}\nabla_\theta^2 \ell(f_{\theta_\mathcal{T}}, \boldsymbol{x}^c)$, as shown in Equation 9. Therefore, the variance of the new synthetic dataset $\mathcal{S}'$ is greater than that of $\mathcal{S}$ without perturbing $\theta_\mathcal{T}$.

3.3 Directed Weight Adjustment on $\theta_\mathcal{T}$

Although perturbing $\theta_\mathcal{T}$ could significantly increase the variance of the synthetic dataset $\mathcal{S}$, undirected random perturbation $\Delta\theta$ can also introduce noise, which in turn degrades the performance. We aim to address this limitation by directing the random perturbation $\Delta\theta$ without introducing noise into $\mathcal{S}$. We propose to obtain directed $\Delta\theta$ by solving the following maximization problem:

$$\widetilde{\Delta\theta} = \arg\max_{\Delta\theta} L_\mathbb{B}\left(f_{\theta_\mathcal{T}+\Delta\theta}\right) \quad \text{where} \quad L_\mathbb{B}\left(f_{\theta_\mathcal{T}+\Delta\theta}\right) = \sum_{\boldsymbol{x}_i \in \mathbb{B}} \ell\left(f_{\theta_\mathcal{T}+\Delta\theta}, \boldsymbol{x}_i\right), \tag{11}$$

where $\mathbb{B} \subset \mathcal{T}$ represents a randomly selected subset of $\mathcal{T}$, and $|\mathbb{B}| \ll |\mathcal{T}|$. As such, $\widetilde{\Delta\theta}$ will not introduce unanticipated noise when synthesizing $\mathcal{S}$. The randomly selected $\mathbb{B}$ ensures that the randomness of $\widetilde{\Delta\theta}$ continues to benefit the diversity of $\mathcal{S}$. Next, we will demonstrate this theoretically.

Effective dataset distillation should provide concise and critical guidance from the original dataset $\mathcal{T}$ when synthesizing the distilled dataset. Here, this guidance is introduced primarily through the converged weight parameters $\theta_\mathcal{T}$, *i.e.*,

$$\theta_\mathcal{T} = \arg\min_{\theta} L_\mathcal{T}\left(f_{\theta_\mathcal{T}}\right) \quad \text{where} \quad L_\mathcal{T}\left(f_{\theta_\mathcal{T}}\right) = \sum_{\boldsymbol{x}_i \in \mathcal{T}} \ell\left(f_{\theta_\mathcal{T}}, \boldsymbol{x}_i\right), \tag{12}$$

where $\theta_\mathcal{T}$ contains informative features of $\mathcal{T}$ because it achieves minimized training loss over $\mathcal{T}$. We demonstrate that $\widetilde{\Delta\theta}$, obtained from Equation 11, decreases the training loss computed over $\mathcal{T} \setminus \mathbb{B}$, which, in fact, highlights the features of $\mathcal{T} \setminus \mathbb{B}$. By applying a first-order Taylor expansion, we obtain:

$$L_{\mathcal{T}\setminus\mathbb{B}}\left(f_{\theta_\mathcal{T}+\widetilde{\Delta\theta}}\right) \approx L_{\mathcal{T}\setminus\mathbb{B}}\left(f_{\theta_\mathcal{T}}\right) + \nabla_\theta L_{\mathcal{T}\setminus\mathbb{B}}\left(f_{\theta_\mathcal{T}}\right)\widetilde{\Delta\theta}. \tag{13}$$

Since $\theta_\mathcal{T}$ is optimized until reaching a local minimum with respect to the loss function computed over the training set $\mathcal{T}$, we have:

$$\nabla_\theta L_\mathcal{T}\left(f_{\theta_\mathcal{T}}\right) = \nabla_\theta L_\mathbb{B}\left(f_{\theta_\mathcal{T}}\right) + \nabla_\theta L_{\mathcal{T}\setminus\mathbb{B}}\left(f_{\theta_\mathcal{T}}\right) = \mathbf{0} \quad \text{thus} \quad \nabla_\theta L_{\mathcal{T}\setminus\mathbb{B}}\left(f_{\theta_\mathcal{T}}\right) = -\nabla_\theta L_\mathbb{B}\left(f_{\theta_\mathcal{T}}\right),$$

where $\mathbf{0}$ is the tensor of zeros with the same dimension as $\theta_\mathcal{T}$. Substitute it back into Equation 13, we have:

$$\begin{aligned}
L_{\mathcal{T}\setminus\mathbb{B}}\left(f_{\theta_\mathcal{T}+\widetilde{\Delta\theta}}\right) - L_{\mathcal{T}\setminus\mathbb{B}}\left(f_{\theta_\mathcal{T}}\right) &\approx \nabla_\theta L_{\mathcal{T}\setminus\mathbb{B}}\left(f_{\theta_\mathcal{T}}\right)\widetilde{\Delta\theta} \\
&= -\nabla_\theta L_\mathbb{B}\left(f_{\theta_\mathcal{T}}\right)\widetilde{\Delta\theta} \\
&\approx -\left(L_\mathbb{B}\left(f_{\theta_\mathcal{T}+\widetilde{\Delta\theta}}\right) - L_\mathbb{B}\left(f_{\theta_\mathcal{T}}\right)\right) \leq 0,
\end{aligned} \tag{14}$$

$L_\mathbb{B}(f_{\theta_\mathcal{T}+\widetilde{\Delta\theta}})$ will clearly be greater than $L_\mathbb{B}(f_{\theta_\mathcal{T}})$, as indicated by Equation 11. Thus, we demonstrate that the directed $\widetilde{\Delta\theta}$ results in less noise and improved performance. In summary, after resolving $\widetilde{\Delta\theta}$ as in Equation 11, our proposed method synthesizes data instance $\boldsymbol{s}_i$ by solving:

$$\tilde{\boldsymbol{s}}_i = \arg\min_{\boldsymbol{s} \in \mathbb{R}^d} \mathcal{L} \quad \text{where} \quad \mathcal{L} = \left[\ell\left(f_{\theta_\mathcal{T}+\widetilde{\Delta\theta}}, \boldsymbol{s}_i\right) + \lambda\mathcal{L}_{\text{mean}}\left(f_{\theta_\mathcal{T}}, \boldsymbol{s}_i\right) + \lambda_{\text{var}}\mathcal{L}_{\text{var}}\left(f_{\theta_\mathcal{T}}, \boldsymbol{s}_i\right)\right]. \tag{15}$$

https://doi.org/10.52202/079017-3796

4 Experiments

To evaluate the effectiveness of the proposed method, we have conducted extensive comparison experiments with SOTA methods on various datasets including CIFAR-10/100 (32×32, 10/100 classes) [16], Tiny-ImageNet (64×64, 200 classes) [18], and ImageNet-1K (224×224, 1000 classes) [3] using diverse network architectures like ResNet-(18, 50, 101) [11], MobileNetV2 [33], ShuffleNetV2 [26], EfficientNet-B0 [37], and VGGNet-16 [35]. We conduct our experiments on the server with one Nvidia Tesla A100 40GB GPU.

Solving $\widetilde{\Delta\theta}$. Before we conduct our experiments, we propose to use a gradient descent approach to solve $\widetilde{\Delta\theta}$ in Equation 11. There are two coefficients, K and ρ, used in the gradient descent approach. K represents the number of steps, and ρ normalizes the magnitude of the directed weight adjustment. The details for solving $\widetilde{\Delta\theta}$ can be found in Line 7 of Algorithm 1.

Experiment Setting. Unless otherwise specified, we default to using ResNet-18 as the backbone for distillation. For ImageNet-1K, we use the pre-trained model provided by Torchvision while for CIFAR-10/100 and Tiny-ImageNet, we modify the original architecture under the suggestion in [10]. More detailed hyper-parameter settings can be found in Appendix A.2.1.

Baselines and Metrics. We conduct comparison with seven Dataset Distillation methods including DC [54], DM [53], CAFE [42], MTT [1], TESLA [2], SRe2L [46], and DataDAM [32]. For all the considered comparison methods, we assess the quality of the distilled dataset by measuring the Top-1 classification accuracy on the original validation set using models trained on them from scratch. Blue cells in all tables highlight the highest performance.

4.1 Results & Discussions

CIFAR-10/100. As shown in Table 1, our DWA exhibits superior performance compared to conventional dataset distillation methods, particularly evident on CIFAR-100 with a larger distillation budget. For instance, our DWA yields over a 10% performance enhancement compared to MTT [1] with $\texttt{ipc} = 50$. Leveraging a more robust distillation backbone like ResNet-18, our approach surpasses the SOTA method SRe2L [46] across all considered settings. Specifically, we achieve more than 5% and 8% accuracy improvement on CIFAR-10 and CIFAR-100, respectively.

Table 1: Comparison with SOTA dataset distillation baselines on CIFAR-10/100. Unless otherwise specified, we use the same network architecture for distillation and validation. Following the settings in their original papers, DC [54], DM [53], CAFE [42], MTT [1], and TESLA [2] use ConvNet-128 (*small model*). For SRe2L [46], ResNet-18 (*large model*) is used for synthesis and validation.

Dataset	ipc	ConvNet						ResNet-18	
		DC [54]	DM [53]	CAFE [42]	MTT [1]	TESLA [2]	DWA (ours)	SRe2L [46]	DWA (ours)
CIFAR-10	10	44.9±0.5	48.9±0.6	46.3±0.6	65.4±0.7	66.4±0.8	45.0±0.4	27.2±0.4	32.6±0.4
	50	53.9±0.5	63.0±0.4	55.5±0.6	71.6±0.7	72.6±0.7	63.3±0.7	47.5±0.5	53.1±0.3
CIFAR-100	10	25.2±0.3	29.7±0.3	27.8±0.3	40.1±0.4	41.7±0.3	47.6±0.4	31.6±0.5	39.6±0.6
	50	-	43.6±0.4	37.9±0.3	47.7±0.2	47.9±0.3	59.0±0.1	52.2±0.3	60.9±0.5

Table 2: Comparison with SOTA dataset distillation baselines on Tiny-ImageNet and ImageNet-1K. Unless otherwise specified, we use the same network architecture for distillation and validation. Following the settings in their original papers, MTT [1], and TESLA [2] use ConvNet-128 (*small model*). For SRe2L [46], ResNet-18 (*large model*) is used for synthesis, and the distilled dataset is evaluated on ResNet-18, 50, and 101. † indicates MTT is performed on a 10-class subset of the full ImageNet-1K dataset.

Dataset	ipc	ConvNet			ResNet-18		ResNet-50		ResNet-101	
		MTT [1]	DataDAM [32]	TESLA [2]	SRe2L [46]	DWA (ours)	SRe2L	DWA (ours)	SRe2L	DWA (ours)
Tiny-ImageNet	50	28.0±0.3	28.7±0.3	-	41.1±0.4	52.8±0.2	42.2±0.5	53.7±0.2	42.5±0.2	54.7±0.3
	100	-	-	-	49.7±0.3	56.0±0.2	51.2±0.4	56.9±0.4	51.5±0.3	57.4±0.3
ImageNet-1K	10	64.0±1.3†	6.3±0.0	17.8±1.3	21.3±0.6	37.9±0.2	28.4±0.1	43.0±0.5	30.9±0.1	46.9±0.4
	50	-	-	27.9±1.2	46.8±0.2	55.2±0.2	55.6±0.3	62.3±0.1	60.8±0.5	63.3±0.7
	100	-	-	-	52.8±0.3	59.2±0.3	61.0±0.4	65.7±0.4	62.8±0.2	66.7±0.2

Figure 2: Visualization of distilled images for the goldfish class. Panels (a) and (b) show the synthesized results by SRe2L [46] and our DWA, respectively. The synthetic data instances generated by our DWA method exhibit significantly greater diversity compared to those produced by SRe2L, highlighting the effectiveness of our approach in capturing a broader range of features.

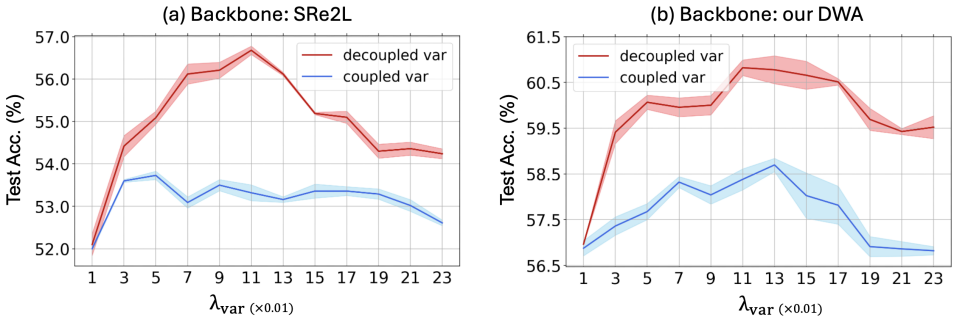

Figure 3: Analysis of decoupled $\mathcal{L}_{\text{var}}$ coefficient. We vary λ_{var} across a wide range of $(0.01 \sim 0.23)$. 'decoupled var' indicates λ_{var} is changing individually with a fixed mean component whose weight defaults to 0.01. 'coupled var' represents the weight of the mean and λ_{var} change in tandem. (a) and (b) illustrate the performance of the original SRe2L [46] and our DWA in these two scenarios, respectively. This analysis is conducted on CIFAR-100 using ResNet-18. Each λ_{var} undergoes five independent experiments, with variance indicated by lighter color shades.

Tiny-ImageNet & ImageNet-1K. Compared with CIFAR-10/100, ImageNet datasets are more closely reflective of real-world scenarios. Table 2 lists the related results. Due to the limited scalability capacity of conventional distillation paradigm, only a few methods have conducted evaluation on ImageNet datasets. Here we provide a comprehensive comparison with SRe2L [46], which has been validated as the most effective one for distilling large-scale dataset. It is obvious that our method significantly outperforms SRe2L on all `ipc` settings and validation models. For instance, our DWA surpasses SRe2L by 16.6% when $\texttt{ipc} = 10$ on ImageNet-1K using ResNet-18. Figure 2 further provides the visualization results, the enhanced diversity is the key driver behind the substantial performance improvement.

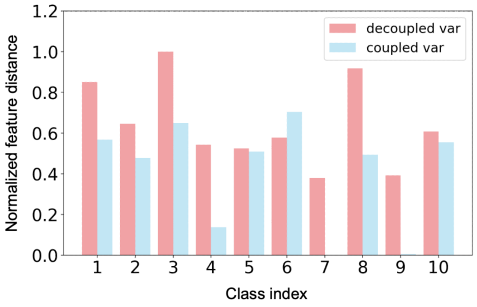

Figure 4: Normalized feature distance of decoupled variance component with $\lambda_{\text{var}} = 0.11$ (the weight of mean component defaults to 0.01) and coupled variance component with $\lambda_{\text{BN}} = 0.11$. ResNet-18's last convolutional layer outputs are used for feature distance calculation (see Appendix A.2.2). Ten classes are randomly chosen from CIFAR-100 distilled dataset.

4.2 Ablation Study

Decoupled $\mathcal{L}_{\text{var}}$ Coefficient. We first test our hypothesis, as outlined in Section 3.1, positing that strengthening $\mathcal{L}_{\text{mean}}$ conflicts with the emphasis on $\mathcal{L}_{\text{var}}$, which is critical for ensuring diversity in synthetic datasets. Therefore, we compare the synthetic dataset distilled with an emphasis on $\mathcal{L}_{\text{BN}}$ (which strengthens both $\mathcal{L}_{\text{mean}}$ and $\mathcal{L}_{\text{var}}$) against one that emphasizes $\mathcal{L}_{\text{var}}$ alone. As depicted in Figure 3, focusing solely on $\mathcal{L}_{\text{var}}$ outperforms the combined emphasis on λ_{BN} in both SRe2L [46] and our proposed Directed Weight Adjustment (DWA). These experimental results verify our hypothesis in Section 3.1, indicating the optimal value of the decoupled coefficient $\mathcal{L}_{\text{var}}$ is 0.11. We also employ

https://doi.org/10.52202/079017-3796

Table 3: An ablation study of DWA was conducted using various network architectures. The synthetic dataset was distilled by ResNet-18 from the CIFAR-100 dataset. We use ✘ to denote the distilled dataset without weight adjustment, ◯ to denote the distilled dataset with random weight adjustment, and ✔ to represent Directed Weight Adjustment (DWA).

	ipc = 10			ipc = 50		
Perturbation	✘	◯	✔	✘	◯	✔
ResNet-18	30.6±0.7	14.9±0.1	39.6±0.6	56.1±0.4	56.2±0.6	60.3±0.5
ResNet-50	26.5±1.1	15.0±0.2	35.2±0.7	55.7±0.9	57.1±0.5	60.6±0.8
MobileNetV2	18.2±0.5	14.4±1.2	27.8±0.7	46.9±0.9	50.7±0.6	53.6±0.2
ShuffleNet	10.3±0.7	10.7±0.1	19.4±0.9	30.9±1.1	39.1±0.1	41.7±0.8
EfficientNet	11.8±0.4	11.1±0.7	20.2±0.4	28.6±1.0	38.8±1.0	40.7±0.3

the normalized feature distance as a metric to comprehensively evaluate our emphasis. This metric measures the mutual feature distances between instances, as defined in Appendix A.2.2. By randomly selecting 10 classes from CIFAR-100, we calculate the normalized feature distances between synthetic datasets emphasized by the decoupled $\mathcal{L}_{\text{var}}$ and the coupled $\mathcal{L}_{\text{BN}}$. The findings, illustrated in Figure 4, validate our hypothesis from a different perspective.

Directed Weight Adjustment. We clarify the necessity of restricting the direction of weight adjustment in Section 3.3. To test its effectiveness, we apply a random $\Delta\theta$, sampled from a Gaussian Distribution, to $\theta_{\mathcal{T}}$. As shown in Table 3, we assess synthetic datasets derived from three scenarios: no weight adjustment, random weight adjustment, and our directed weight adjustment (DWA) method, using the CIFAR-100 dataset. The results, examined across various architectures, underscore the importance of directing weight adjustments in distillation processes. Notably, we observe performance degradation in the synthetic dataset optimized with random weight adjustment at $\texttt{ipc} = 10$ compared to those without weight adjustment. This decline occurs because, at smaller ipc values, the noise introduced by random weight adjustment outweighs the benefits of diversity. However, as the number of synthetic instances increases, diversity becomes more effective in capturing a broader range of features, leading to improved performance, as reflected at $\texttt{ipc} = 50$.

Table 4: Cross-architecture performance of distilled dataset of CIFAR-100 using ResNet-18 and ConvNet-128.

	ipc	Methods	MobileNetv2	ShuffleNet	EfficientNet	VGG-16	ResNet-50	ConvNet-128
ResNet-18	10	SRe2L	16.1±0.5	11.8±0.7	11.1±0.3	19.2±0.2	22.4±1.3	19.4±0.2
		DWA (ours)	27.8±0.7	19.4±0.9	20.2±0.4	30.0±0.5	35.2±0.7	27.3±0.3
	50	SRe2L	43.2±0.2	27.5±1.1	24.9±1.7	40.4±1.2	52.8±0.7	19.4±0.2
		DWA (ours)	53.6±0.2	41.7±0.8	40.7±0.3	51.6±0.4	60.6±0.8	37.0±0.3
ConvNet-128	10	SRe2L	28.7±1.3	25.3±0.4	18.0±0.9	21.5±1.6	41.8±0.2	-
		DWA (ours)	37.3±0.1	25.3±0.4	24.5±0.4	29.6±1.3	47.1±0.3	47.6±0.4
	50	SRe2L	48.8±0.4	49.3±0.7	45.7±0.8	38.9±0.5	53.4±0.5	-
		DWA (ours)	53.5±0.3	44.37±0.4	45.7±0.8	38.9±0.5	56.3±0.3	59.0±0.1

Parameters Study on K and ρ. Apart from direction, the number of steps K and magnitude ρ of perturbation also influence the distillation process. Figure 5 illustrates the grid search for these two hyper-parameters and demonstrates the positive impact of perturbation, which is achieved effortlessly, requiring no meticulous manual parameter tuning. In our experiments, we set $K = 12$ and $\rho = 15e^{-3}$ for all the datasets. Readers can adjust these hyper-parameters according to their specific circumstances (different datasets and networks) to obtain better results.

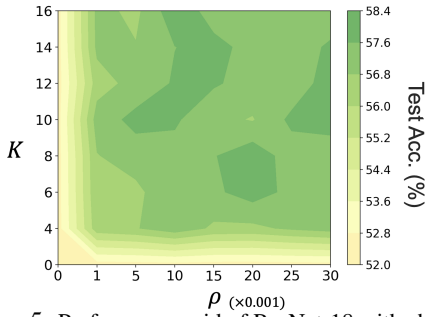

Figure 5: Performance grid of ResNet-18 with changes in perturbation steps K and magnitude ρ.

Cross-Architecture Generalization. The generalizability across different architectures is a key feature for assessing the effectiveness of the distilled dataset. In this section, we evaluate the surrogate dataset condensed by different backbones (ResNet-18 and ConvNet-128) on various archi-

tectures including MobileNetV2 [33], ShuffleNetV2 [26], EfficientNet-B0 [37], and VGGNet-16 [35]. The experimental results are reported in Table 4 and Table 5. It is evident that our DWA-synthesized dataset can effectively generalize across various architectures. Notably, for $\texttt{ipc} = 50$ on CIFAR-100 with ShuffleNetV2, EfficientNet-B0, and ConvNet-128—three architectures not involved in the data synthesis phase—our method achieves impressive classification performance, with accuracies of 41.7%, 40.7%, and 37.0%, respectively, outperforming the latest SOTA method, SRe2L [46], by 14.2%, 15.8%, and 17.6%. In Appendix A.2.3, we further extend the proposed method to a vision transformer-based model, DeiT-Tiny [40].

5 Related Works

Dataset Distillation [43] emerges as a derivative of Knowledge Distillation (KD) [9], emphasizing data-centric efficiency over traditional model-centric one. Previous studies have explored various strategies to condense datasets, including performance matching, gradient matching [54, 52, 19] distribution matching [42, 53, 55, 48, 4], and trajectory matching [1, 2, 5, 6, 21, 41].

Table 5: Cross-architecture performance of distilled dataset of ImageNet-1K using ResNet-18.

ipc	Methods	MobileNetv2	ShuffleNet	EfficientNet
10	SRe2L	15.4±0.2	9.0±0.7	11.7±0.2
	DWA (ours)	29.1±0.3	11.4±0.6	37.4±0.5
50	SRe2L	48.3±0.5	9.0±0.6	53.6±0.4
	DWA (ours)	51.6±0.5	28.5±0.5	56.3±0.4

What distinguishes DD from KD is the bi-level optimization, which considers both model parameters and image pixels. The consequent complexity and computational burden intricate optimization significantly diminish the effectiveness of the aforementioned methods. To address this issue, SRe2L [46] introduced a three-step paradigm known as *Squeeze-Recover-Relabel*. This approach relies on the highly encoded distribution prior, *i.e.*, the running mean and running variance in the BN layer, to circumvent supervision provided by model training. With this decoupled optimization, SRe2L is able to extend DD to high-resolution and large-scale datasets like ImageNet-1K.

Another critical challenge in dataset compression, not limited to distillation, is how to represent the original dataset distribution with a scarcity of synthetic data samples [36]. Previous research claims that the diversity of a dataset can be evaluated by spatial distribution [27], the maximum dispersion or convex hull volume [47], and coverage [56]. Conventional dataset distillation [49, 15] treats the synthetic compact dataset as an integrated optimizable tensor without specialized guarantees for diversity and relies entirely on the matching objectives mentioned above. Recognizing this limitation, Dream [23] proposed using cluster centers to induce synthesis and ensure adequate diversity. Besides, SRe2L resorts to the second-order statistics, *i.e.*, variance of representations in pre-trained weights to provide diversity.

6 Conclusion

In this work, we hypothesize that ensuring diversity is crucial for effective dataset distillation. Our findings indicate that the random initialization of synthetic data instances contributes minimally to ensuring that each instance captures unique knowledge from the original dataset. We validate our hypothesis through both theoretical and empirical approaches, demonstrating that enhancing diversity significantly benefits dataset distillation. To this end, we propose a novel method, Directed Weight Adjustment (DWA), which introduces diversity in synthesis by customizing weight adjustments for each mini-batch of synthetic data. This approach ensures that each mini-batch condenses a variety of knowledge. Extensive experiments, particularly on the large-scale ImageNet-1K dataset, confirm the superior performance of our proposed DWA method.

Limitations and Future work. While DWA provides a straightforward and efficient approach to introducing diversity in dataset distillation, its reliance on the sampling of a random distribution to adjust weight parameters presents limitations. Increasing the variance of the random distribution can introduce unexpected noise, thereby bottlenecking overall performance. Future investigations could explore synthesizing data instances in a sequential manner, encouraging later instances to consciously distinguish themselves from earlier ones, thereby further enhancing diversity.

https://doi.org/10.52202/079017-3796

Acknowledgements

This research is supported by Jiawei Du's A*STAR Career Development Fund (CDF) C233312004 and Joey Tianyi Zhou's A*STAR SERC Central Research Fund (Use-inspired Basic Research). This research is also supported by National Natural Science Foundation of China under Grant 62301213.

References

[1] George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A. Efros, and Jun-Yan Zhu. Dataset distillation by matching training trajectories. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 10708–10717, 2022.

[2] Justin Cui, Ruochen Wang, Si Si, and Cho-Jui Hsieh. Scaling up dataset distillation to imagenet-1k with constant memory. In *Proc. Int. Conf. Mach. Learn. (ICML)*, pages 6565–6590, 2023.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 248–255, 2009.

[4] Wenxiao Deng, Wenbin Li, Tianyu Ding, Lei Wang, Hongguang Zhang, Kuihua Huang, Jing Huo, and Yang Gao. Exploiting inter-sample and inter-feature relations in dataset distillation. *arXiv preprint arXiv:2404.00563*, 2024.

[5] Jiawei Du, Yidi Jiang, Vincent Y. F. Tan, Joey Tianyi Zhou, and Haizhou Li. Minimizing the accumulated trajectory error to improve dataset distillation. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 3749–3758, 2023.

[6] Jiawei Du, Qin Shi, and Joey Tianyi Zhou. Sequential subset matching for dataset distillation. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2023.

[7] Yunzhen Feng, Shanmukha Ramakrishna Vedantam, and Julia Kempe. Embarrassingly simple dataset distillation. In *Adv. Neural Inf. Process. Syst. Workshop (NeurIPS Workshop)*, 2023.

[8] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2021.

[9] Jianping Gou, Baosheng Yu, Stephen J. Maybank, and Dacheng Tao. Knowledge distillation: A survey. *Int. J. Comput. Vis.*, 129(6):1789–1819, 2021.

[10] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 9726–9735, 2020.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 770–778, 2016.

[12] Muyang He, Yexin Liu, Boya Wu, Jianhao Yuan, Yueze Wang, Tiejun Huang, and Bo Zhao. Efficient multimodal learning from data-centric perspective. *arXiv preprint arXiv:2402.11530*, 2024.

[13] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

[14] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[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 *Proc. Int. Conf. Mach. Learn. (ICML)*, pages 11102–11118, 2022.

[16] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[17] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper R. R. Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, and Vittorio Ferrari. The open images dataset V4: unified image classification, object detection, and visual relationship detection at scale. *Int. J. Comput. Vis. (IJCV)*, 128(7):1956–1981, 2020.

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

[19] Saehyung Lee, Sanghyuk Chun, Sangwon Jung, Sangdoo Yun, and Sungroh Yoon. Dataset condensation with contrastive signals. In *Proc. Int. Conf. Mach. Learn. (ICML)*, pages 12352–12364, 2022.

[20] Shiye Lei and Dacheng Tao. A comprehensive survey of dataset distillation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(1):17–32, 2024.

[21] Dai Liu, Jindong Gu, Hu Cao, Carsten Trinitis, and Martin Schulz. Dataset distillation by automatic training trajectories. *arXiv preprint arXiv:2407.14245*, 2024.

[22] Songhua Liu and Xinchao Wang. MGDD: A meta generator for fast dataset distillation. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2023.

[23] Yanqing Liu, Jianyang Gu, Kai Wang, Zheng Zhu, Wei Jiang, and Yang You. DREAM: efficient dataset distillation by representative matching. In *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 17268–17278. IEEE, 2023.

[24] Noel Loo, Ramin Hasani, Mathias Lechner, and Daniela Rus. Dataset distillation with convexified implicit gradients. In *International Conference on Machine Learning*, pages 22649–22674. PMLR, 2023.

[25] Noel Loo, Ramin M. Hasani, Alexander Amini, and Daniela Rus. Efficient dataset distillation using random feature approximation. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2022.

[26] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet V2: practical guidelines for efficient CNN architecture design. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pages 122–138, 2018.

[27] Adyasha Maharana, Prateek Yadav, and Mohit Bansal. D2 pruning: Message passing for balancing diversity and difficulty in data pruning. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2024.

[28] Timothy Nguyen, Zhourong Chen, and Jaehoon Lee. Dataset meta-learning from kernel ridge-regression. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021.

[29] Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 20596–20607, 2021.

[30] Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. Gdumb: A simple approach that questions our progress in continual learning. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pages 524–540. Springer, 2020.

[31] Noveen Sachdeva and Julian J. McAuley. Data distillation: A survey. *Trans. Mach. Learn. Res.*, 2023.

[32] Ahmad Sajedi, Samir Khaki, Ehsan Amjadian, Lucy Z. Liu, Yuri A. Lawryshyn, and Konstantinos N. Plataniotis. Datadam: Efficient dataset distillation with attention matching. In *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 17051–17061, 2023.

 https://doi.org/10.52202/079017-3796

[33] Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 4510–4520, 2018.

[34] Yuzhang Shang, Zhihang Yuan, and Yan Yan. MIM4DD: mutual information maximization for dataset distillation. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2023.

[35] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015.

[36] Peng Sun, Bei Shi, Daiwei Yu, and Tao Lin. On the diversity and realism of distilled dataset: An efficient dataset distillation paradigm. *arXiv preprint arXiv:2312.03526*, 2023.

[37] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *Proc. Int. Conf. Mach. Learn. (ICML)*, pages 6105–6114, 2019.

[38] Kushal Tirumala, Daniel Simig, Armen Aghajanyan, and Ari Morcos. D4: improving LLM pretraining via document de-duplication and diversification. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2023.

[39] 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 *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019.

[40] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021.

[41] Kai Wang, Zekai Li, Zhi-Qi Cheng, Samir Khaki, Ahmad Sajedi, Ramakrishna Vedantam, Konstantinos N Plataniotis, Alexander Hauptmann, and Yang You. Emphasizing discriminative features for dataset distillation in complex scenarios. *arXiv preprint arXiv:2410.17193*, 2024.

[42] 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 *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 12186–12195, 2022.

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

[44] Xilie Xu, Jingfeng Zhang, Feng Liu, Masashi Sugiyama, and Mohan S. Kankanhalli. Efficient adversarial contrastive learning via robustness-aware coreset selection. 2024.

[45] Hongxu Yin, Pavlo Molchanov, José M. Álvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K. Jha, and Jan Kautz. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 8712–8721, 2020.

[46] Zeyuan Yin, Eric P. Xing, and Zhiqiang Shen. Squeeze, recover and relabel: Dataset condensation at imagenet scale from A new perspective. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2023.

[47] Yu Yu, Shahram Khadivi, and Jia Xu. Can data diversity enhance learning generalization? In *Proc. Int. Conf. Comput. Linguistics (COLING)*, pages 4933–4945, 2022.

[48] Hansong Zhang, Shikun Li, Pengju Wang, Dan Zeng, and Shiming Ge. Echo: Efficient dataset condensation by higher-order distribution alignment. *arXiv preprint arXiv:2312.15927*, 2023.

[49] Lei Zhang, Jie Zhang, Bowen Lei, Subhabrata Mukherjee, Xiang Pan, Bo Zhao, Caiwen Ding, Yao Li, and Dongkuan Xu. Accelerating dataset distillation via model augmentation. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 11950–11959, 2023.

[50] Xin Zhang, Jiawei Du, Yunsong Li, Weiying Xie, and Joey Tianyi Zhou. Spanning training progress: Temporal dual-depth scoring (TDDS) for enhanced dataset pruning. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2024.

[51] Xin Zhang, Jiawei Du, Ping Liu, and Joey Tianyi Zhou. Breaking class barriers: Efficient dataset distillation via inter-class feature compensator. *arXiv preprint arXiv:2408.06927*, 2024.

[52] Bo Zhao and Hakan Bilen. Dataset condensation with differentiable siamese augmentation. In *Proc. Int. Conf. Mach. Learn. (ICML)*, pages 12674–12685, 2021.

[53] Bo Zhao and Hakan Bilen. Dataset condensation with distribution matching. In *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, pages 6503–6512, 2023.

[54] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. Dataset condensation with gradient matching. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021.

[55] Ganlong Zhao, Guanbin Li, Yipeng Qin, and Yizhou Yu. Improved distribution matching for dataset condensation. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 7856–7865, 2023.

[56] Haizhong Zheng, Rui Liu, Fan Lai, and Atul Prakash. Coverage-centric coreset selection for high pruning rates. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2023.

[57] Muxin Zhou, Zeyuan Yin, Shitong Shao, and Zhiqiang Shen. Self-supervised dataset distillation: A good compression is all you need. *arXiv preprint arXiv:2404.07976*, 2024.

 https://doi.org/10.52202/079017-3796

A Appendix

A.1 Minimizing $\mathcal{L}_{\text{mean}}$ and $\mathcal{L}_{\text{var}}$ can be contradictory

To prove that minimizing $\mathcal{L}_{\text{mean}}$ and $\mathcal{L}_{\text{var}}$ can result in contradictory objectives for some existing instances, we will demonstrate that the gradients required to minimize $\mathcal{L}_{\text{mean}}$ and $\mathcal{L}_{\text{var}}$, respectively, may point in opposite directions. Specifically, for any arbitrary instance $\boldsymbol{s}_i \in \mathcal{S}$, our goal is to establish:

$$\frac{\partial \mathcal{L}_{\text{mean}}}{\partial \boldsymbol{s}_i} \cdot \frac{\partial \mathcal{L}_{\text{var}}}{\partial \boldsymbol{s}_i} < 0, \tag{16}$$

For $\frac{\partial \mathcal{L}_{\text{mean}}}{\partial \boldsymbol{s}_i}$, we have

$$\begin{aligned} \frac{\partial \mathcal{L}_{\text{mean}}}{\partial \boldsymbol{s}_i} &= \frac{\partial \left[\mu(\mathcal{S}) - \mu\left(\mathcal{T}\right)\right]^2}{\partial \boldsymbol{s}_i} = \frac{\partial \left[\mu\left(\mathcal{S}\right) - \mu\left(\mathcal{T}\right)\right]^2}{\partial \mu\left(\mathcal{S}\right)} \cdot \frac{\partial \mu\left(\mathcal{S}\right)}{\partial \boldsymbol{s}_i} \\ &= 2\left[\mu\left(\mathcal{S}\right) - \mu\left(\mathcal{T}\right)\right] \cdot \frac{1}{|\mathcal{S}|}, \end{aligned} \tag{17}$$

because $\mu(\mathcal{S}) = \frac{1}{|\mathcal{S}|}\boldsymbol{s}_i + \sum_{j \neq i} \frac{1}{|\mathcal{S}|}\boldsymbol{s}_j$, thus $\frac{\partial \mu(\mathcal{S})}{\partial \boldsymbol{s}_i} = \frac{1}{|\mathcal{S}|}$. For $\frac{\partial \mathcal{L}_{\text{var}}}{\partial \boldsymbol{s}_i}$, we have

$$\begin{aligned} \frac{\partial \mathcal{L}_{\text{var}}}{\partial \boldsymbol{s}_i} &= \frac{\partial \left[\sigma^2\left(\mathcal{S}\right) - \sigma^2\left(\mathcal{T}\right)\right]^2}{\partial \boldsymbol{s}_i} = \frac{\partial \left[\sigma^2\left(\mathcal{S}\right) - \sigma^2\left(\mathcal{T}\right)\right]^2}{\partial \sigma^2\left(\mathcal{S}\right)} \cdot \frac{\partial \sigma^2\left(\mathcal{S}\right)}{\partial \boldsymbol{s}_i} \\ &= 2\left[\sigma^2\left(\mathcal{S}\right) - \sigma^2\left(\mathcal{T}\right)\right] \cdot \frac{\partial \sigma^2\left(\mathcal{S}\right)}{\partial \boldsymbol{s}_i} \\ &= 2\left[\sigma^2\left(\mathcal{S}\right) - \sigma^2\left(\mathcal{T}\right)\right] \cdot \frac{\partial \left[\frac{1}{|\mathcal{S}|}\left(\boldsymbol{s}_i - \mu\left(\mathcal{S}\right)\right)^2 + \sum_{j \neq i} \frac{1}{|\mathcal{S}|}\left(\boldsymbol{s}_j - \mu\left(\mathcal{S}\right)\right)^2\right]}{\partial \boldsymbol{s}_i} \\ &= 2\left[\sigma^2\left(\mathcal{S}\right) - \sigma^2\left(\mathcal{T}\right)\right] \cdot \frac{1}{|\mathcal{S}|} \frac{\partial \left(\boldsymbol{s}_i - \mu\left(\mathcal{S}\right)\right)^2}{\partial \boldsymbol{s}_i} \\ &= 2\left[\sigma^2\left(\mathcal{S}\right) - \sigma^2\left(\mathcal{T}\right)\right] \cdot \frac{1}{|\mathcal{S}|} \cdot 2\left(\boldsymbol{s}_i - \mu\left(\mathcal{S}\right)\right) \cdot \frac{\partial \left(\boldsymbol{s}_i - \mu\left(\mathcal{S}\right)\right)}{\partial \boldsymbol{s}_i} \\ &= 2\left[\sigma^2\left(\mathcal{S}\right) - \sigma^2\left(\mathcal{T}\right)\right] \cdot \frac{1}{|\mathcal{S}|} \cdot 2\left(\boldsymbol{s}_i - \mu\left(\mathcal{S}\right)\right) \cdot \left(1 - \frac{1}{|\mathcal{S}|}\right). \end{aligned} \tag{18}$$

Substitute Equation 17 and Equation 18 back into Equation 16,

$$\begin{aligned} &\frac{\partial \mathcal{L}_{\text{mean}}}{\partial \boldsymbol{s}_i} \cdot \frac{\partial \mathcal{L}_{\text{var}}}{\partial \boldsymbol{s}_i} \\ =&2\left[\mu\left(\mathcal{S}\right) - \mu\left(\mathcal{T}\right)\right] \cdot \frac{1}{|\mathcal{S}|} \cdot 2\left[\sigma^2\left(\mathcal{S}\right) - \sigma^2\left(\mathcal{T}\right)\right] \cdot \frac{1}{|\mathcal{S}|} \cdot 2(\boldsymbol{s}_i - \mu(\mathcal{S})) \cdot (1 - \frac{1}{|\mathcal{S}|}) \\ =&\left[\frac{2}{|\mathcal{S}|}\right]^3 (|\mathcal{S}| - 1)\left[\mu\left(\mathcal{S}\right) - \mu\left(\mathcal{T}\right)\right] \cdot \left[\sigma^2\left(\mathcal{S}\right) - \sigma^2\left(\mathcal{T}\right)\right] \cdot \left(\boldsymbol{s}_i - \mu\left(\mathcal{S}\right)\right), \end{aligned} \tag{19}$$

Let $R = [\mu(\mathcal{S}) - \mu(\mathcal{T})] \cdot [\sigma^2(\mathcal{S}) - \sigma^2(\mathcal{T})]$, where R is a constant that can be either positive or negative, depending on the values of $\mu(\mathcal{S}), \mu(\mathcal{T}), \sigma^2(\mathcal{S})$, and $\sigma^2(\mathcal{T})$. Suppose $R > 0$. In this scenario, instances for which $(\boldsymbol{s}_i - \mu(\mathcal{S})) < 0$ will encounter contradictory objectives in optimization. Conversely, if $R < 0$, instances where $(\boldsymbol{s}_i - \mu(\mathcal{S})) > 0$ will face similar contradictions.

A.2 Experiments

A.2.1 Hyper-parameter Settings

Table 6, Table 7, and Table 8 list the hyper-parameter settings of our method on experimental datasets. We maintain consistency with SRe2L for a fair comparison.

Table 6: Hyper-parameter settings for CIFAR-10/100.

Distillation		**Validation**	
#Iteration	1000	#Epoch	400
Batch Size	100	Batch Size	128
Optimizer	Adam with $\{\beta_1, \beta_2\} = \{0.5, 0.9\}$	Optimizer	AdamW with weight decay of 0.01
Learning Rate	0.25 using cosine decay	Learning Rate	0.001 using cosine decay
Augmentation	-	Augmentation	`RandomCrop` `RandomHorizontalFlip`
λ_{var}	11	Tempreture	30
ρ, K	$15e^{-3}, 12$		

Table 7: Hyper-parameter settings for Tiny-ImageNet.

Distillation		**Validation**	
#Iteration	2000	#Epoch	200
Batch Size	100	Batch Size	128
Optimizer	Adam with $\{\beta_1, \beta_2\} = \{0.5, 0.9\}$	Optimizer	SGD with weight decay of 0.9
Learning Rate	0.1 using cosine decay	Learning Rate	0.2 using cosine decay
Augmentation	`RandomResizedCrop` `RandomHorizontalFlip`	Augmentation	`RandomResizedCrop` `RandomHorizontalFlip`
λ_{var}	11	Tempreture	20
ρ, K	$15e^{-3}, 12$		

Table 8: Hyper-parameter settings for ImageNet-1K.

Distillation		**Validation**	
#Iteration	2000	#Epoch	300
Batch Size	100	Batch Size	128
Optimizer	Adam with $\{\beta_1, \beta_2\} = \{0.5, 0.9\}$	Optimizer	AdamW with weight decay of 0.01
Learning Rate	0.25 using cosine decay	Learning Rate	0.001 using cosine decay
Augmentation	`RandomResizedCrop` `RandomHorizontalFlip`	Augmentation	`RandomResizedCrop` `RandomHorizontalFlip`
λ_{var}	2	Tempreture	20
ρ, K	$15e^{-3}, 12$		

A.2.2 Feature Distance Calculation

In Figure 4, we use feature distance $\mathcal{D}_{fea}$ to measure the diversity of distilled dataset. The following is how the class-wise feature distance is calculated,

$$\mathcal{D}_{fea}^{c} = \sum_{i=1}^{\texttt{ipc}} \sum_{j=1}^{\texttt{ipc}} \| g_{\theta_\mathcal{T}}(\tilde{\boldsymbol{s}}_i^c) - g_{\theta_\mathcal{T}}(\tilde{\boldsymbol{s}}_j^c) \|^2, \tag{20}$$

where $g_{\theta_\mathcal{T}}(\tilde{\boldsymbol{s}}_i^c)$ and $g_{\theta_\mathcal{T}}(\tilde{\boldsymbol{s}}_j^c)$ are the latent representations of i-th and j-th synthetic instances of class c, specifically the outputs from the last convolutional layer.

A.2.3 Generalization to Vision Transformer-based Models

We acknowledge that our proposed approach cannot be directly applied to models without BN layers, such as Vision Transformers (ViTs). Our baseline solution, SRe2L, involves developing a ViT-BN model that replaces all LayerNorm layers with BN layers and adds additional BN layers

https://doi.org/10.52202/079017-3796

between the two linear layers of the feed-forward network. We followed their solution and conducted cross-architecture experiments with DeiT-Tiny [40] on the ImageNet-1K dataset. The results are listed in Table 9. The results demonstrate that our approach can be applied to ViT-BN with superior performance compared to the baseline.

Table 9: Generalization to a vision transformer-based model DeiT-Tiny.

	Methods	DeiT-Tiny	ResNet-18	ResNet-50	ResNet-101
ResNet-18	SRe2L	15.41	46.80	55.60	60.81
	DWA (ours)	22.72	55.20	62.30	63.3
DeiT-Tiny-BN	SRe2L	25.36	24.69	31.15	33.16
	DWA (ours)	37.0	32.64	40.77	43.15

A.2.4 Application to Downstream Tasks

We evaluate our proposed DWA on a continual learning task, based on an effective continual learning method GDumb [30]. Class-incremental learning was performed under strict memory constraints on the CIFAR-100 dataset, with 20 images per class (`ipc` $= 20$). CIFAR-100 was divided into five tasks, and a ConvNet was trained on our distilled dataset, with accuracy measured as new classes were incrementally introduced. As shown in Table 10, DWA significantly outperforms SRe2L across all class-incremental stages, demonstrating superior retention of knowledge throughout the learning process.

Table 10: Application to continual learning task.

Class	20	40	60	80	100
SRe2L	15.7	10.6	9.0	7.9	6.9
DWA (ours)	34.6	25.7	22.5	20.2	18.1

A.2.5 Computational Overhead of Distillation

We compare the average time required to generate one `ipc` using ResNet-18 on CIFAR-100. As shown in Table 11, our proposed DWA incurs only a 7.32% increase in computational overhead while significantly enhancing the diversity of the synthetic dataset. This additional overhead arises from the K-step directed weight perturbation applied before generating each `ipc`, as detailed in lines 6-7 of Algorithm 1,

$$\text{For } k = 1 \text{ to } K \text{ do}$$
$$\Delta\theta_k = \Delta\theta_{k-1} + \frac{\rho}{K}\nabla L_{\mathcal{S}_0^i}\left(f_{\theta_T+\Delta\theta_{k-1}}\right).$$

Since each `ipc` requires 1000 iterations of forward-backward propagation for generation, the additional $K = 12$ forward-backward propagations required by DWA are negligible in the overall distillation process.

Table 11: Computational overhead of distillation on CIFAR-100 with ResNet-18.

Methods	Avg. time for generating one `ipc`
SRe2L	116.58 s (100%)
DWA (ours)	125.12 s (107.32%)

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims in the abstract and introduction accurately reflect the paper's contributions and scope. The abstract and introduction summarize the key contributions and findings, which are consistently supported by detailed methodologies, experiments, and results in the main body. The claims are accurately presented without exaggeration.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper does discuss its limitations, adhering to guidelines regarding assumptions, scope of claims, performance factors, computational efficiency, privacy, fairness, and honesty.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

https://doi.org/10.52202/079017-3796

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The paper ensures that for each theoretical result, it provides the full set of assumptions and a complete (and correct) proof. This adheres to the guidelines by clearly stating or referencing all assumptions in the statement of theorems, numbering and cross-referencing all theorems, formulas, and proofs, and providing formal proofs either in the main paper or supplemental material. Additionally, the paper appropriately references any external theorems or lemmas relied upon in the proofs.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper fully discloses all the information needed to reproduce the main experimental results, ensuring transparency and reproducibility. This includes detailed descriptions of experimental setups, methodologies, and any necessary parameters or configurations. The paper offers clear instructions and explanations on how to replicate the results.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: This paper provides implementation details and algorithm descriptions for reproduction. We also release our codes for reproduction in camera-ready version.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies all the training and test details necessary to reproduce the results, including datasets, hyperparameters, optimizer type, and how they were chosen.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

https://doi.org/10.52202/079017-3796

Answer: [Yes]

Justification: Our experiments were rigorously conducted with five repetitions each, and we meticulously reported both the mean values and standard deviations for each experimental trial.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper comprehensively details compute resources in both the experiments section and supplementary materials, covering GPU type, memory, and storage specifics.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: The research aligns with the NeurIPS Code of Ethics, ensuring ethical standards are upheld throughout the study.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: Our work is foundational research, and therefore, it does not have direct societal impacts to discuss.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper does not involve the release of data or models that have a high risk for misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

 https://doi.org/10.52202/079017-3796

Answer: [Yes]

Justification: The paper provides proper credit to asset creators, citing relevant papers and explicitly mentioning license and terms of use. URLs are included where possible, and all licenses are respected.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The new assets introduced in the paper are well-documented, providing comprehensive details alongside the assets, including training procedures, licenses, limitations, and consent processes, ensuring transparency and reproducibility.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

https://doi.org/10.52202/079017-3796