Diversity-Enhanced Distribution Alignment for Dataset Distillation

Hongcheng Li[1,2,3] Yucan Zhou[4*] Xiaoyan Gu[1,2,3†] Bo Li[1,2,3] Weiping Wang[1]
[1]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3]State Key Laboratory of Cyberspace Security Defense, Beijing, China
[4]College of Intelligence and Computing, Tianjin University, Tianjin, China
{lihongcheng,guxiaoyan,libo,wangweiping}@iie.ac.cn, zhouyucan@tju.edu.cn

Abstract

Dataset distillation, which compresses large-scale datasets into compact synthetic representations (i.e., distilled datasets), has become crucial for the efficient training of modern deep learning architectures. While existing large-scale dataset distillation methods leverage a pre-trained model through batch normalization statistics alignment, they neglect the essential role of covariance matrices in preserving inter-feature correlations, resulting in reduced diversity in the distilled datasets. In this paper, we propose a simple yet effective approach, Diversity-Enhanced Distribution Alignment (DEDA), which enhances the diversity of distilled data by leveraging inter-feature relationships. Our method first establishes Gaussian distribution alignment by matching the means and covariances of each class in the original dataset with those of the distilled dataset in the feature space of a pre-trained model. Since features within the last layer of a pre-trained model are often highly similar within each class, aligning distributions in this layer cannot obtain diversified distilled data, resulting in gradient starvation during downstream training tasks. To overcome this limitation, we introduce a regularizer that constrains the covariance matrix of the distilled data in the last layer to maximize diagonal elements while minimizing non-diagonal elements. Extensive evaluations across CIFAR-10/100, Tiny-ImageNet, and ImageNet-1K demonstrate state-of-the-art performance without additional computational overhead.

1. Introduction

The exponential growth of large-scale datasets has driven significant advancements in deep learning, enabling deep neural networks to achieve dramatic success across various tasks [7, 13, 18, 23, 38]. However, the increasing storage requirements and extended training time due to the expanding dataset size present a fundamental challenge. Dataset distillation (or condensation) has emerged as a crucial solution, which compresses the large-scale datasets into significantly smaller and more representative ones (i.e., distilled datasets) [40, 44]. Such distilled datasets can accelerate model training in downstream tasks, while maintaining the training performance of the original dataset across a wide range of model architectures.

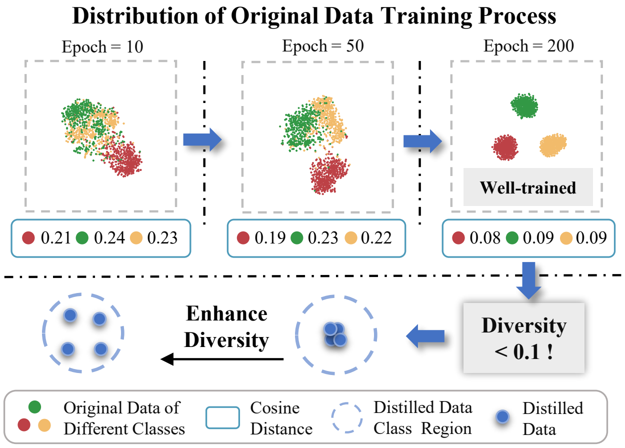


Figure 1. Visualization of the last layer feature distribution during CIFAR-100 model training. As training progresses, features become increasingly compact. In a well-trained model, samples within each class exhibit high cosine similarity (cosine distance < 0.1). This compact feature space can limit the diversity of distilled data when aligning with the original data features.

Although many dataset distillation methods have shown strong performance on moderate-scale benchmarks, such as CIFAR-10/100 [11, 21, 25, 42], their scalability to large-scale datasets like ImageNet-1K remains inherently constrained. One primary bottleneck is the need to load the original data during each optimization step of the distilled data, resulting in significant memory and computational

[*] Corresponding author. Email: zhouyucan@tju.edu.cn
[†] Corresponding author. Email: guxiaoyan@iie.ac.cn

overhead. To address these challenges, SRe2L [43] introduces a novel decoupled framework for efficient large-scale dataset distillation. Instead of loading the original data repeatedly, this approach first trains a well-converged model on the original dataset. Then, the distilled samples are optimized by aligning their embeddings in the Batch Normalization (BN) layer with the statistics of the BN layer in the pre-trained well-converged model. While aligning global BN statistics effectively preserves the overall information of the original dataset, it will make all distilled samples capture highly similar information, leading to high inter-class similarity. To mitigate this issue, LPLD [41] proposes aligning intra-class BN statistics to enhance the class discriminability of the distilled data. However, we argue that simply aligning the mean and variance of the BN layer's multi-dimensional feature distributions is insufficient to capture the distribution of the original data.

Notably, the covariance matrix is another critical statistic that captures intra-class feature variations, which contains more information than the variance with each non-diagonal element represents the relationship between two feature dimensions. An intuitive way is to align the distilled data with both the mean and covariance of the feature representations from the pre-trained model. However, samples in the last layer of a pre-trained well-converged model tend to be remarkably similar. As illustrated in Figure 1, the distribution of the original samples from the same category in the last-layer is gradually shrunk during the training process. It is evident that as the model converges, samples of a category in the last-layer become increasingly concentrated, with the cosine distance between individual samples and their class centers typically falling below 0.1. As a result, even if the distilled data is aligned with the distribution of the original sample in the last-layer of the pre-trained model, their diversity is still limited, leading to gradient starvation for the model training in downstream tasks.

To enhance the diversity of distilled data, we propose a simple yet effective Diversity-Enhanced Distribution Alignment (DEDA) framework, which preserves the semantic richness of the original dataset through feature distribution alignment. Unlike the batch normalization-based alignment methods, our approach employs Gaussian distribution matching between the original and distilled data in the feature space. To obtain the Gaussian distribution of the original dataset, we design an Offline Gaussian Distribution Estimation, which extracts feature statistics with the pre-trained model by loading the original data once and stores the mean and covariance matrices in a memory bank. We then optimize the distilled data by aligning their feature distributions with the class-specific statistics of the original data in the memory bank to achieve efficient class-wise distillation. Furthermore, to address distribution compactness observed in the last-layer of the pre-trained model, we introduce a covariance regularization. This regularization simultaneously maximizes the diagonal elements to increase feature variance and minimizes the off-diagonal elements to eliminate redundant correlations. It effectively mitigates gradient starvation in downstream tasks caused by the limited samples' diversity in the abstract feature space. Experimental results on various datasets (e.g., CIFAR-10/100, Tiny-ImageNet, and ImageNet-1K) demonstrate that our DEDA can effectively enhance semantic diversity and significantly improve the performance in downstream tasks. The main contributions of this work are as follows:

- We propose a Diversity-Enhanced Distribution Alignment (DEDA) for dataset distillation. To the best of our knowledge, this is the first work to maintain the inter-feature relationships in large-scale dataset distillation to enhance the diversity of the distilled data.
- Due to the high similarity of samples from the same category in the last layer of a well-converged model, we propose a covariance regularization for distilled data to avoid gradient starvation in downstream tasks.
- Our experiments demonstrate that the proposed DEDA achieves state-of-the-art results on multiple benchmarks while maintaining computational efficiency. Moreover, because of the high semantic diversity, our distilled data exhibits superior cross-architecture performance.

2. Related Work

Existing dataset distillation methods can be categorized into two principal paradigms: optimization process alignment and feature space alignment.

Optimization Process Alignment. The optimization process alignment methods aim to incorporate the intermediate training dynamics of the original dataset into the distilled data synthesis process. The foundational framework involves gradient matching [16, 20, 47, 48], which minimizes the distance between the gradients generated by the original and distilled datasets during model parameter updates. Subsequently, trajectory alignment methods [3, 4, 8, 24] ensure consistency across entire optimization paths rather than at each individual optimization step. However, the information learned by the model from the original data at different training stages is inconsistent, and therefore, indiscriminate matching can lead to a loss of information in the distilled data. To address this, SeqMatch [9] and DATM [14] enhance the process by explicitly dividing the model into different training stages for alignment, thereby improving the diversity of the information captured in the distilled data. Moreover, DREAM [26] further improves the quality of the distilled data by dynamically selecting samples through clustering original data representatives. While these approaches achieve notable performance improvements, they are computationally expensive due to the iterative model updating and data synthesis processes.

Feature Space Alignment. To reduce computational costs, this paradigm directly aligns the feature distributions between the original and distilled data, without requiring model updates [49]. CAFE [39] aligns the prototypes of intermediate features from deep networks. To reduce feature redundancy, DataDAM [31] utilizes spatial attention mechanisms to align key features. Additionally, M3D [46] reformulates prototype matching within Reproducing Kernel Hilbert Spaces (RKHS). However, these methods typically use randomly initialized models to extract features, which can lead to inaccurate embeddings. To enhance embedding accuracy, IDM [50] and DANCE [45] propose to use multiple pre-trained feature extractors for more robust feature extraction. While pre-trained models improve embedding accuracy, only matching class prototypes will limit the diversity of the distilled data. IID [6] and DSDM [22] address this limitation by aligning both the prototypes and covariance matrices of class features, thereby preserving semantic richness. However, the joint processing of original and distilled data introduces significant memory demands.

Large-Scale Dataset Distillation. To achieve data distillation on large-scale datasets, some methods have introduced pre-trained generative models for data synthesis, where their performance relies heavily on the pre-trained models [12, 35]. Orthogonal to the generation-base methods, recent advances address scalability bottlenecks through decoupled strategies [36, 42, 43]. The pioneering SRe2L framework [43] introduces a decoupled distillation paradigm that first trains a well-converged model on the original data and then optimizes the distilled data by aligning local batch statistics with the global Batch Normalization (BN) parameters of the pre-trained model. However, this global BN alignment mechanism enforces a uniform optimization objective across all synthetic samples, inherently limiting inter-class discrimination of the distilled data. LPLD [41] attempts to enhance class discriminability by aligning intra-class BN statistics. However, capturing a pre-trained model's BN statistics (mean and variance) is insufficient to guarantee the diversity of the distilled data. To address this limitation, alternative strategies have emerged with different alignment mechanisms. G-VBSM [33] matches multiple statistical characteristics of different network architectures. Meanwhile, DWA [10] introduces training perturbations and variance-weighted adjustments to enhance sample diversity. However, these methods incur extra computational overhead due to additional model training. Instead, we directly leverage the covariance matrix to capture inter-feature relationships, effectively enhancing the diversity without additional computational costs.

3. Preliminary

Background. In this paper, we denote the original large-scale dataset as $\mathcal{T} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{|\mathcal{T}|}$, where $y_i \in \mathcal{Y} = \{0, 1, \ldots, C-1\}$ represents the corresponding label. The distilled dataset is represented as $\mathcal{S} = \{(\boldsymbol{s}_i, y_i)\}_{i=1}^{|\mathcal{S}|}$, with each class containing IPC (Images Per Class) samples. The total size of the distilled dataset is $|\mathcal{S}| = \text{IPC} \times C \ll |\mathcal{T}|$, ensuring that the distilled dataset is significantly smaller than the original dataset. The objective of dataset distillation is to compress a large-scale training dataset $\mathcal{T}$ into a much smaller distilled dataset $\mathcal{S}$ while maintaining model performance. Specifically, a model $\mathcal{M}_{\boldsymbol{\theta}_\mathcal{S}}$ trained on the distilled dataset $\mathcal{S}$ should achieve comparable performance to a model $\mathcal{M}_{\boldsymbol{\theta}_\mathcal{T}}$ trained on the full dataset $\mathcal{T}$ when evaluated on a test set. This objective can be formally expressed as:

$$\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{test}}\left[\ell\left(\mathcal{M}_{\boldsymbol{\theta}_\mathcal{T}}, \boldsymbol{x}\right)\right] \approx \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{test}}\left[\ell\left(\mathcal{M}_{\boldsymbol{\theta}_\mathcal{S}}, \boldsymbol{x}\right)\right], \quad (1)$$

where $\ell(\cdot, \boldsymbol{x})$ denotes the loss function that measures the model's performance on a test sample $\boldsymbol{x}$, and $\mathcal{D}_{test}$ represents the test set.

Large-Scale Dataset Distillation. Classical dataset distillation methods have demonstrated success in compressing information for medium-sized datasets [3, 31, 47], but they struggle to scale effectively to large-scale datasets. To overcome this limitation, SRe2L [43] introduces a decoupled learning paradigm, consisting of three key stages: Squeeze, Recover, and Relabel. In the Squeeze stage, a pre-trained model $\boldsymbol{\theta}_\mathcal{T}$ is first trained on the original dataset $\mathcal{T}$, thereby compressing the feature of $\mathcal{T}$ into $\boldsymbol{\theta}_\mathcal{T}$. In the recover stage, the pre-trained model $\boldsymbol{\theta}_\mathcal{T}$ is then used to optimize distilled data $\mathcal{S}$. The optimization objective is to align $\mathcal{S}$ with the information encoded in $\boldsymbol{\theta}_\mathcal{T}$. Specifically, the goal is to minimize the following loss function:

$$\arg\min_{\mathcal{S}} \left[\mathcal{L}_{\text{CE}}(\boldsymbol{\theta}_\mathcal{T}, \mathcal{S}) + \mathcal{L}_{\text{BN}}(\boldsymbol{\theta}_\mathcal{T}, \mathcal{S})\right], \quad (2)$$

where $\mathcal{L}_{\text{CE}}$ is the standard cross-entropy loss function, and $\mathcal{L}_{\text{BN}}$ is the batch normalization (BN) alignment loss. The BN loss $\mathcal{L}_{\text{BN}}$ ensures that the mean and variance of the normalized feature distribution of $\mathcal{S}$ are aligned with those stored in the batch normalization layer of $\boldsymbol{\theta}_\mathcal{T}$. Finally, in the Relabel stage, the downstream training task on $\mathcal{S}$ is performed through soft label alignment, where soft labels are generated from the logits of $\boldsymbol{\theta}_\mathcal{T}$.

4. Method

We illustrate the framework of our proposed Diversity-Enhanced Distribution Alignment for Dataset Distillation in Figure 2. In the first stage, we estimate the Gaussian distribution for each class c in the original dataset $\mathcal{T}$ using features extracted for the intermediate layer l of the pre-trained model $\boldsymbol{\theta}_\mathcal{T}$, i.e., $\mathcal{N}(\boldsymbol{\mu}^l_{\mathcal{T},c}, \boldsymbol{Cov}^l_{\mathcal{T},c})$. These statistics of the Gaussian distribution in each layer are then stored in a memory bank. In the second stage, during the optimization process of the distilled data $\mathcal{S}_c$, we retrieve the corresponding statistics from the memory bank and feed them

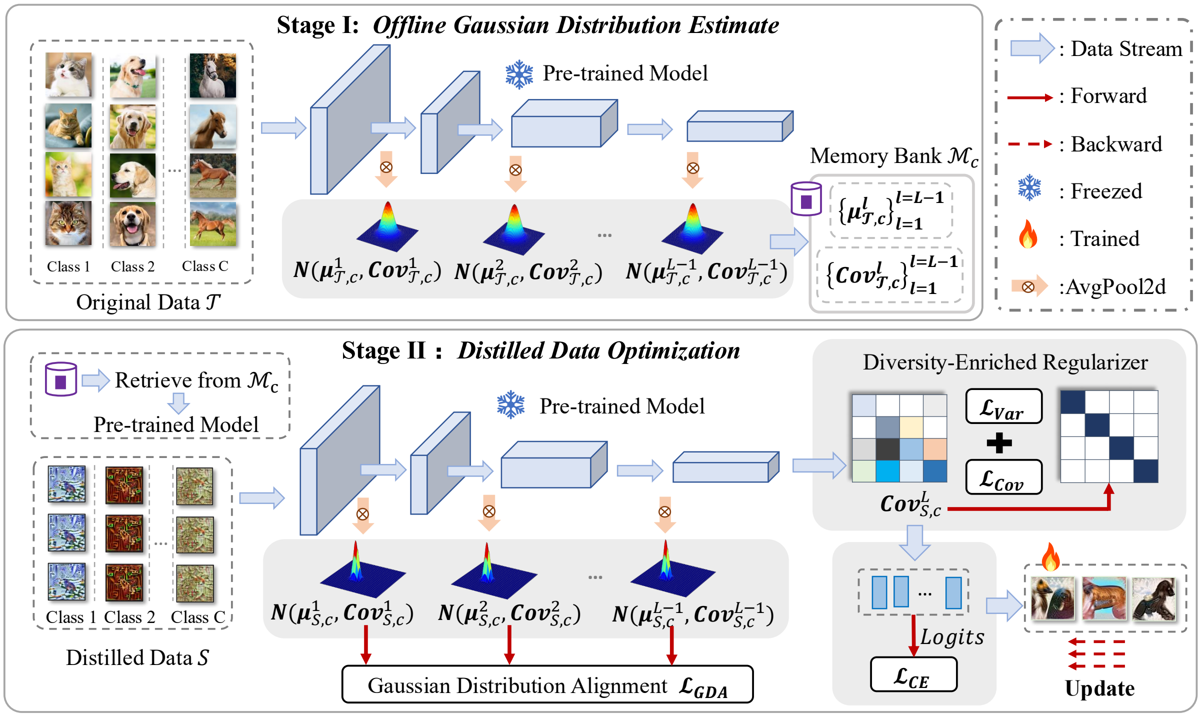

Stage I: Offline Gaussian Distribution Estimate
Class 1
Class 2
Class C
Original Data $\mathcal{T}$
Pre-trained Model
Memory Bank $\mathcal{M}_c$
$\{\boldsymbol{\mu}^l_{\mathcal{T},c}\}_{l=1}^{l=L-1}$
$\{\boldsymbol{Cov}^l_{\mathcal{T},c}\}_{l=1}^{l=L-1}$
$N(\boldsymbol{\mu}^1_{\mathcal{T},c}, \boldsymbol{Cov}^1_{\mathcal{T},c})$
$N(\boldsymbol{\mu}^2_{\mathcal{T},c}, \boldsymbol{Cov}^2_{\mathcal{T},c})$
...
$N(\boldsymbol{\mu}^{L-1}_{\mathcal{T},c}, \boldsymbol{Cov}^{L-1}_{\mathcal{T},c})$
: Data Stream
: Forward
: Backward
: Freezed
: Trained
:AvgPool2d
Stage II : Distilled Data Optimization
Retrieve from $\mathcal{M}_c$
Pre-trained Model
Class 1
Class 2
Class C
Distilled Data $\mathcal{S}$
Pre-trained Model
$N(\boldsymbol{\mu}^1_{\mathcal{S},c}, \boldsymbol{Cov}^1_{\mathcal{S},c})$
$N(\boldsymbol{\mu}^2_{\mathcal{S},c}, \boldsymbol{Cov}^2_{\mathcal{S},c})$
...
$N(\boldsymbol{\mu}^{L-1}_{\mathcal{S},c}, \boldsymbol{Cov}^{L-1}_{\mathcal{S},c})$
Gaussian Distribution Alignment $\mathcal{L}_{GDA}$
Diversity-Enriched Regularizer
$\mathcal{L}_{Var}$
$\mathcal{L}_{Cov}$
$\boldsymbol{Cov}^L_{\mathcal{S},c}$
Logits
$\mathcal{L}_{CE}$
Update

Figure 2. Illustration of our proposed DEDA. Stage I: Offline Gaussian Distribution Estimate. Pre-trained model statistics (mean $\boldsymbol{\mu}^l_{\mathcal{T},c}$ and covariance $\boldsymbol{Cov}^l_{\mathcal{T},c}$) across network layers and stored in a memory bank $\mathcal{M}_c$. Stage II: Distilled Data Optimization. DEDA (1) retrieves target statistics from $\mathcal{M}_c$, (2) computes three objective components: Gaussian distribution alignment ($\mathcal{L}_{\text{GDA}}$), diversity-enriched regularizer ($\mathcal{L}_{\text{Var}}$, $\mathcal{L}_{\text{Cov}}$), and standard cross-entropy loss ($\mathcal{L}_{\text{CE}}$), and (3) updates distilled data through optimization iterations.

to $\boldsymbol{\theta}_{\mathcal{T}}$. To ensure consistency across layers, we align the feature distribution $\mathcal{N}(\boldsymbol{\mu}^l_{\mathcal{S},c}, \boldsymbol{Cov}^l_{\mathcal{S},c})$ of $\mathcal{S}_c$ with the corresponding statistics from $\boldsymbol{\theta}_{\mathcal{T}}$. To address the issue of feature space compactness in the last layer of $\boldsymbol{\theta}_{\mathcal{T}}$, we introduce an explicit regularization term to constrain the covariance matrix of the distilled data $\boldsymbol{Cov}^L_{\mathcal{S},c}$. Specifically, we maximize the diagonal elements (variance loss $\mathcal{L}_{\text{Var}}$) and minimize the non-diagonal elements of the covariance matrix (covariance loss $\mathcal{L}_{\text{Cov}}$), which encourages a maximally separable feature distribution for the distilled samples. The overall loss function is summarized in Section 4.3.

4.1. Gaussian Distribution Alignment

Although previous work [42, 43] that aligns the mean and variance of batch normalization (BN) layers in a pre-trained model using Eq. (2) has achieved promising results, we believe that these statistical measures are insufficient to fully capture the distribution of the original data. We argue that covariance is also a crucial statistical component, as it encapsulates the relationships between feature dimensions. Therefore, we propose an offline Gaussian distribution estimation to better describe the diversity of the original data.

Specifically, for each class $c \in \{1, \dots, C\}$, we model its sample distribution at layer l as a Gaussian distribution. For the samples in class class c, we extract their features by feeding them into the pre-trained model $\boldsymbol{\theta}_{\mathcal{T}}$, producing feature tensors $\boldsymbol{f}^l(\boldsymbol{x})$ at each intermediate layer. Given the high dimensionality of the feature tensor (e.g., the first layer of ResNet-18 has a dimensionality of $64 \times 56 \times 56$), directly computing the covariance matrix is computationally expensive. To reduce this cost while preserving critical information, we apply an average pooling operation:

$$\tilde{\boldsymbol{f}}^l(\boldsymbol{x}) = \text{AvgPool2d}\left(\boldsymbol{f}^l(\boldsymbol{x})\right), \tag{3}$$

where $\boldsymbol{f}^l(\boldsymbol{x}) \in \mathbb{R}^{D_l \times H_l \times W_l}$, with D_l indicating the number of channels in layer l, and H_l and W_l denoting the height and width of the feature maps in layer l, respectively. After average pooling, the feature dimension is reduced to $\tilde{\boldsymbol{f}}^l(\boldsymbol{x}) \in \mathbb{R}^{D_l}$. Using the pooled features, we compute the feature statistics for each class c, specifically the mean vector $\boldsymbol{\mu}^l_{\mathcal{T},c}$ and the covariance matrix $\boldsymbol{Cov}^l_{\mathcal{T},c}$ are:

$$\boldsymbol{\mu}^l_{\mathcal{T},c} = \frac{1}{|\mathcal{T}_c|} \sum_{i=1}^{|\mathcal{T}_c|} \tilde{\boldsymbol{f}}^l(\boldsymbol{x}_i), \tag{4}$$

$$\boldsymbol{Cov}^l_{\mathcal{T},c} = \tfrac{1}{|\mathcal{T}_c|} \textstyle\sum_{i=1}^{|\mathcal{T}_c|} (\tilde{\boldsymbol{f}}^l(\boldsymbol{x}_i) - \boldsymbol{\mu}^l_{\mathcal{T},c})(\tilde{\boldsymbol{f}}^l(\boldsymbol{x}_i) - \boldsymbol{\mu}^l_{\mathcal{T},c})^\top. \tag{5}$$

where $\boldsymbol{\mu}^l_{\mathcal{T},c} \in \mathbb{R}^{D_l}$ and $\boldsymbol{Cov}^l_{\mathcal{T},c} \in \mathbb{R}^{D_l \times D_l}$. To efficiently manage the the Gaussian distribution of each class, we construct a memory bank to the mean and covariance of each

Gaussian distribution. This memory bank allows for fast access during the optimization process, improving computational efficiency.

During the optimization process of the distilled data, we sequentially optimize the distilled data for each class c, denoted as $\mathcal{S}_c$. Before optimizing $\mathcal{S}_c$, we retrieve the mean and covariance of the Gaussian distribution statistics for class c from the memory bank and assign them to $\boldsymbol{\theta}_{\mathcal{T}}$. Subsequently, following the same approach as in the original data modeling, we model the Gaussian distribution for $\mathcal{S}_c$ to obtain $\mathcal{N}(\boldsymbol{\mu}^l_{\mathcal{S},c}, \boldsymbol{Cov}^l_{\mathcal{S},c})$ and align these statistics with those of the original data in $\boldsymbol{\theta}_{\mathcal{T}}$. Based on this, we define the Gaussian Distribution Alignment (GDA) loss for $L-1$ layers (excluding the last layer) as follows:

$$\mathcal{L}_{\text{GDA}} = \sum_{l=1}^{L-1} \left(\|\boldsymbol{\mu}^l_{\mathcal{S},c} - \boldsymbol{\mu}^l_{\mathcal{T},c}\|_2^2 + \gamma \|\boldsymbol{Cov}^l_{\mathcal{S},c} - \boldsymbol{Cov}^l_{\mathcal{T},c}\|_2^2 \right), \tag{6}$$

where γ is a hyperparameter that balances the contributions of the mean and covariance alignment terms.

4.2. Diversity-Enriched Regularizer

While layer-wise Gaussian distribution alignment in a pre-trained model effectively preserves the diversity of the original data, the excessively compact and homogeneous features in the last layer L lead to a constrained matching space. Consequently, aligning with the last-layer features of the pre-trained model may result in insufficient diversity of the distilled data, leading to gradient starvation [1, 2, 29, 30] for model training in downstream tasks.

To address this challenge, we propose an explicit covariance regularization for the last-layer features of the distilled data. For the covariance matrix of the distilled data's last layer, denoted as $\boldsymbol{Cov}^L_{\mathcal{S},c} \in \mathbb{R}^{D_L \times D_L}$, we introduce two complementary properties to enhance feature representation: (1) Maximization of diagonal variances $\boldsymbol{\sigma} = \text{Diag}(\boldsymbol{Cov}^L_{\mathcal{S},c}) \in \mathbb{R}^{D_L}$ to ensure comprehensive utilization of each feature dimension, and (2) Minimization of non-diagonal covariances to promote feature independence and reduce redundancy. Therefore, our proposed regularizer consists of two components:

$$\mathcal{L}_{\text{Var}} = \frac{1}{D_L} \sum_{i=1}^{D_L} \max\left(0, 1 - \sqrt{\boldsymbol{\sigma}_i}\right), \tag{7}$$

$$\mathcal{L}_{\text{Cov}} = \frac{1}{D_L(D_L-1)} \sum_{i \neq j} \left| \boldsymbol{Cov}^L_{\mathcal{S},c} \right|^2_{i,j}. \tag{8}$$

The variance regularization term $\mathcal{L}_{\text{Var}}$ employs a hinge-based formulation to enhance feature diversity, while the covariance term $\mathcal{L}_{\text{Cov}}$ acts as a cross-correlation suppression regularizer aiming to improve representational efficiency. Further theoretical analysis is provided in Appendix 9.

Algorithm 1: DEDA

```
Input: $\mathcal{T}$: Original training dataset; $\mathcal{S}$: Distilled
       synthetic dataset; $T$: Number of training
       iterations; $\boldsymbol{\theta}_{\mathcal{T}}$: Pre-trained model; $C$:
       Number of classes; $\eta$: Learning rate.
Initialize distilled dataset $\mathcal{S} \leftarrow \emptyset$
Stage I: Offline Gaussian Distribution Estimate
Group samples by class $c$ to form subset $\mathcal{T}_c \subseteq \mathcal{T}$.
Compute feature mean $\{\boldsymbol{\mu}^l_{\mathcal{T},c}\}_{l=1}^{l=L-1}$ via Eq. (4).
Compute covariance $\{\boldsymbol{Cov}^l_{\mathcal{T},c}\}_{l=1}^{l=L-1}$ via Eq. (5).
Store $\{\boldsymbol{\mu}^l_{\mathcal{T},c}\}_{l=1}^{l=L-1}$ and $\{\boldsymbol{Cov}^l_{\mathcal{T},c}\}_{l=1}^{l=L-1}$ in
  memory bank $\mathcal{M}_c$.
Stage II: Distilled Data Optimization
for each class $c = 1$ to $C$ do
    Initialize the model parameters using the stored
      statistics from the memory bank $\mathcal{M}c$:
      $\boldsymbol{\theta}_{\mathcal{T}} \leftarrow \{\boldsymbol{\mu}^l_{\mathcal{T},c}\}_{l=1}^{l=L-1}, \{\boldsymbol{Cov}^l_{\mathcal{T},c}\}_{l=1}^{l=L-1}$.
    for iteration $i = 1$ to $T$ do
        Compute $\mathcal{L}_{\text{GDA}}$ via Eq.(6).
        Compute $\mathcal{L}_{\text{Var}}$ via Eq.(7).
        Compute $\mathcal{L}_{\text{Cov}}$ via Eq.(8).
        Compute the total loss $\mathcal{L}$ via Eq.(9).
        Update $\mathcal{S}_c \leftarrow \mathcal{S}_c - \eta \nabla_{\mathcal{S}_c} \mathcal{L}$.
Output: Distilled synthetic dataset $\mathcal{S}$
```

4.3. Overall Loss and Training Algorithm

Following previous work [10, 41, 43], we preserve the cross-entropy loss $\mathcal{L}_{\text{CE}}$ from Eq.(2) to maintain intra-class discriminability. In addition, our composite loss function comprises three key components: the Gaussian distribution alignment loss $\mathcal{L}_{\text{GDA}}$, variance regularization loss $\mathcal{L}_{\text{Var}}$, and covariance regularization loss $\mathcal{L}_{\text{Cov}}$. The overall loss function is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{GDA}} + \lambda_1 \mathcal{L}_{\text{Var}} + \lambda_2 \mathcal{L}_{\text{Cov}}, \tag{9}$$

where λ_1 and λ_2 are weighting coefficients. The training procedure is detailed in Algorithm 1.

Computational Complexity. Traditional methods align feature maps across all BN layers by matching their means and variances, leading to a computational complexity of $O(2 \times D_l \times H_l \times W_l)$. In contrast, our DEDA uses spatial average pooling to reduce spatial dimensions and aligns both the means and covariance matrices, lowering the computational complexity to $O(D_l^2 + D_l)$.

5. Experiment

5.1. Experimental Setup

Datasets. We evaluate our dataset distillation method on four widely-used image benchmarks: (1) Low-resolution

Table 1. Comparison of different methods on CIFAR-10, CIFAR-100, and Tiny-ImageNet. For classical dataset distillation methods, ConvNet represents methods using ConvNet-128, with underlined results indicating their best performance. For modern large-scale dataset distillation methods, RN-18 represents methods using ResNet-18, where **bold** results highlight the overall best performance.

	Method	Venue	CIFAR-10		CIFAR-100		Tiny-ImageNet		
			IPC = 10	IPC = 50	IPC = 10	IPC = 50	IPC = 10	IPC = 50	IPC = 100
ConvNet	Random	-	26.0 ± 1.2	43.4 ± 1.0	14.6 ± 0.5	30.0 ± 0.4	5.0 ± 0.2	15.0 ± 0.4	-
	KIP [28]	ICLR21	62.7 ± 0.3	68.6 ± 0.2	28.3 ± 0.1	-	-	-	-
	DC [47]	ICLR21	44.9 ± 0.5	53.9 ± 0.5	32.3 ± 0.3	42.8 ± 0.4	-	-	-
	CAFE [39]	CVPR22	50.9 ± 0.5	62.3 ± 0.4	31.5 ± 0.2	42.9 ± 0.2	-	-	-
	DM [49]	WACV23	48.9 ± 0.6	63.0 ± 0.4	29.7 ± 0.3	43.6 ± 0.4	12.9 ± 0.4	24.1 ± 0.3	-
	MTT [3]	CVPR23	65.3 ± 0.7	71.6 ± 0.2	40.1 ± 0.4	47.7 ± 0.2	23.2 ± 0.2	28.0 ± 0.3	-
	DataDAM [31]	ICCV23	54.2 ± 0.8	67.0 ± 0.4	34.8 ± 0.5	49.4 ± 0.3	18.7 ± 0.3	28.7 ± 0.3	-
RN-18	SRe2L [43]	NeurIPS23	27.2 ± 0.4	47.5 ± 0.5	31.6 ± 0.5	52.2 ± 0.3	16.1 ± 0.2	41.1 ± 0.4	49.7 ± 0.3
	LPLD [41]	NeurIPS24	-	-	-	-	-	48.8 ± 0.4	53.6 ± 0.3
	DEDA	-	$\mathbf{38.2 \pm 0.4}$	$\mathbf{60.2 \pm 0.5}$	$\mathbf{43.6 \pm 0.4}$	$\mathbf{65.1 \pm 0.4}$	$\mathbf{44.5 \pm 0.6}$	$\mathbf{55.2 \pm 0.3}$	$\mathbf{59.3 \pm 0.3}$

Table 2. Comparison on ImageNet-1K using ResNet-{18, 50, 101} . **Bold** results indicate the best results. † denotes the reported results.

Method	ResNet-18			ResNet-50			ResNet-101		
	IPC = 10	IPC = 50	IPC = 100	IPC = 10	IPC = 50	IPC = 100	IPC = 10	IPC = 50	IPC = 100
SRe2L [43]	21.3 ± 0.6	46.8 ± 0.2	52.8 ± 0.3	28.4 ± 0.1	55.6 ± 0.3	61.0 ± 0.4	30.9 ± 0.1	60.8 ± 0.5	62.8 ± 0.2
LPLD [41]	34.6 ± 0.9	55.4 ± 0.3	$\mathbf{59.4 \pm 0.2}$	$41.7^\dagger$	$62.2^\dagger$	$\mathbf{65.7}^\dagger$	-	-	-
DEDA	$\mathbf{36.4 \pm 0.7}$	$\mathbf{56.3 \pm 0.4}$	58.4 ± 0.3	$\mathbf{42.5 \pm 0.5}$	$\mathbf{62.7 \pm 0.4}$	65.4 ± 0.3	$\mathbf{46.3 \pm 0.5}$	$\mathbf{62.4 \pm 0.6}$	$\mathbf{64.9 \pm 0.3}$

datasets: CIFAR-10/100 [17] (32×32 resolution, 60,000 images, 10/100 classes) and Tiny-ImageNet [19] (64×64 resolution, 100,000 images, 200 classes); (2) High-resolution dataset: ImageNet-1K [5] (224×224 resolution, 1.28 million images, 1,000 classes). These datasets vary in complexity, where ImageNet-1K is the most complex due to its high resolution and large image volume.

Implementation Details. We implement our method following the experimental settings established in prior works [41, 43]. Specifically, we use ResNet-18 [15] as the default distillation architecture. We set the covariance matrix hyperparameter to $\gamma = 50$ by default to promote feature diversity while aligning Gaussian distributions. Our regularization strategy uses weighted coefficients of $\lambda_1 = 0.2$ for variance regularization and $\lambda_2 = 4.0$ for covariance regularization. More implementation details are in Section 8.

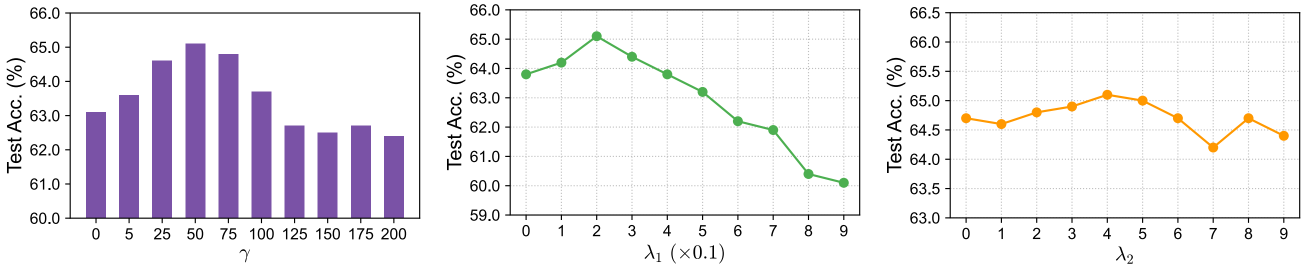
Evaluation Metric. We evaluate the quality of the distilled data using the Top-1 test accuracy on the original dataset's test set. Following the evaluation strategy [41, 43], we use the soft labels generated by the pre-trained teacher model on the distilled data as the ground truth for training a model from scratch. These soft labels are dynamically updated by the teacher model at each validation epoch.

Compared Methods. We evaluate our DEDA method against several state-of-the-art baselines. In addition to random sample selection, we compare with classical dataset distillation methods, including DC [47], KIP [28], DM [49], CAFE [39], MTT [3], and DataDAM [31]. For large-scale dataset distillation, we choose SRe2L [43] and LPLD [41] for comparison. Notably, G-VBSM [33] and DWA [10] are not compared, as they incorporate multiple model training.

5.2. Main Results

CIFAR-10/100 and Tiny-ImageNet. Table 1 presents a comparative analysis of our DEDA framework against state-of-the-art methods on CIFAR-10, CIFAR-100, and Tiny-ImageNet. Our findings highlight two key insights: (1) When utilizing larger pre-trained models (e.g., ResNet-18), DEDA achieves substantial performance gains, outperforming SRe2L by 12.7% on CIFAR-10, 12.9% on CIFAR-100, and 14.1% on Tiny-ImageNet with IPC = 50. These significant improvements validate the effectiveness of our Gaussian distribution matching mechanism in capturing essential features. (2) While conventional ConvNet-128-based methods, such as MTT [3], leverage bilevel optimization and are considered state-of-the-art for small-scale dataset distillation, they struggle to scale effectively to larger datasets. In contrast, DEDA consistently outperforms these approaches on more challenging benchmarks, demonstrating its superior ability to handle dataset expansion, particularly on CIFAR-100 and Tiny-ImageNet.

ImageNet-1K. Table 2 presents a comprehensive evaluation of our DEDA on ImageNet-1K using ResNet-18, ResNet-50, and ResNet-101 architectures. The results re-



(a) Effect of γ.

(b) Effect of λ_1.

(c) Effect of λ_2.

Figure 3. Hyperparameter ablation (γ, λ_1, λ_2) on CIFAR-100 with IPC=50. γ controls the covariance-to-mean ratio in $\mathcal{L}_{\text{GDA}}$, while λ_1 and λ_2 are the weights that are the variance maximization and off-diagonal covariance terms, respectively.

Table 3. Cross-architecture performance of distilled dataset of CIFAR-100 using ResNet-18.

IPC	NetWorks	SRe2L	Ours
10	ResNet-50	22.4 ± 1.3	**39.4 ± 0.9**
	MobileNetV2	16.1 ± 0.5	**30.1 ± 1.1**
	EfficientNetB0	11.1 ± 0.3	**24.6 ± 0.5**
	ShuffleNetV2	11.8 ± 0.7	**23.1 ± 0.8**
	VGG-16	19.2 ± 0.2	**29.7 ± 0.4**
50	ResNet-50	52.8 ± 0.7	**63.7 ± 0.6**
	MobileNetV2	43.2 ± 0.2	**55.6 ± 0.6**
	EfficientNetB0	24.9 ± 1.7	**43.7 ± 1.2**
	ShuffleNetV2	27.5 ± 1.1	**45.1 ± 0.9**
	VGG-16	40.4 ± 1.2	**52.3 ± 0.8**

veal the following insights: (1) Both LPLD and our DEDA outperform SRe2L, highlighting the effectiveness of class-specific alignment in reducing inter-class similarity and enhancing class discriminability. (2) Our DEDA further surpasses LPLD, demonstrating that the introduction of covariance alignment and regularization significantly enhances the diversity of distilled data, ultimately leading to superior performance on downstream tasks.

5.3. Cross-Architecture Evaluation

A more critical evaluation metric for distilled data is its ability to generalize across different network architectures. As shown in Table 3, we comprehensively assess the cross-architecture generalization capability of our CIFAR-100 distilled dataset (trained on ResNet-18) by testing its performance on several unseen network architectures trained from scratch. The evaluation includes seven distinct architectures: ResNet-50 [15], MobileNetV2 [32], EfficientNetB0 [37], ShuffleNetV2 [27], and VGG-16 [34], showing that our method can achieve generalization without the need to optimize distilled data across multiple architectures.

Table 4. Ablation study on CIFAR-100 with IPC=10/50.

Method	$\mathcal{L}_{\text{GDA}}$		DER		IPC	
	MA	CA	$\mathcal{L}_{\text{Var}}$	$\mathcal{L}_{\text{Cov}}$	10	50
SRe2L	-	-	-	-	32.0	47.5
Ours	✓	-	-	-	39.8	63.1
	✓	✓	-	-	41.6	63.8
	✓	✓	✓	-	42.2	64.7
	✓	✓	-	✓	42.1	64.2
	✓	✓	✓	✓	43.6	65.1

5.4. Analysis

Ablation Study. To evaluate the effectiveness of each component in our DEDA, we conduct ablation experiments on CIFAR-100, considering two primary elements: Gaussian Distribution Alignment Loss ($\mathcal{L}_{\text{GDA}}$) and Diversity-Enriched Regularizer (DER). As detailed in Table 4, our analysis reveals four key findings: (1) Our novel approach of storing complete Gaussian statistics (mean and covariance) from the pre-trained model demonstrates superior effectiveness compared to existing methods. The Mean Alignment (MA) component alone outperforms SRe2L's BN-layer alignment by 15.6%, demonstrating the importance of class-specific alignment. (2) The Covariance Alignment (CA) component in $\mathcal{L}_{\text{GDA}}$ provides an additional performance improvement when combined with MA. This demonstrates that preserving inter-feature relationships through covariance matching is crucial for maintaining the diversity of feature dimensions. (3) Our proposed Variance Maximization Regularization ($\mathcal{L}_{\text{Var}}$) applied to last-layer features ensures sufficient diversity in semantic dimensions, yielding further accuracy gains of 0.6% and 0.9%. This validates our hypothesis that feature space expansion promotes better knowledge representation. (4) The Off-diagonal Covariance Minimization Regularization ($\mathcal{L}_{\text{Cov}}$) effectively reduces inter-dimensional redundancy, achieving additional performance improvements of 1.4%

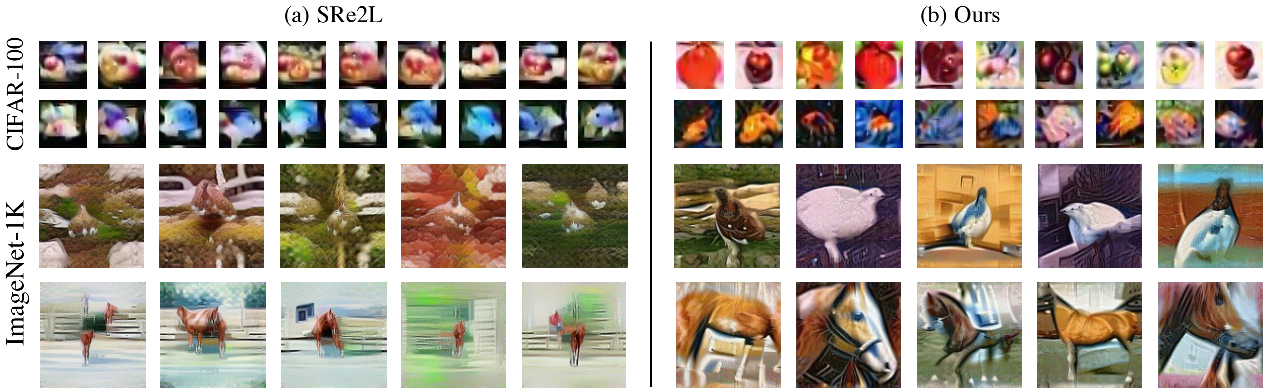


Figure 4. Visualization of distilled data. (a) and (b) show distilled images from SRe2L and our DEDA, respectively. The first two rows display apples and aquarium fish from CIFAR-100, while the last two rows show ptarmigan birds and sorrel horses from ImageNet-1K. Compared to distilled images generated by SRe2L, those produced by our method exhibit significantly greater diversity.

and 0.4%. This confirms that promoting orthogonality in the last-layer feature dimensions effectively reduces the impact of feature redundancy.

Effect of γ. The hyperparameter γ represents the ratio between covariance and mean alignment in our framework. Covariance matrices play a pivotal role as statistical descriptors that encode both intra-class variations and inter-dimensional correlations among feature dimensions. As demonstrated in Figure 3a for CIFAR-100 with IPC=50, we observe substantial performance gains when increasing γ from 0 to 50. However, extending γ beyond this optimal range (50 to 200) leads to performance degradation. This phenomenon stems from an inherent trade-off between covariance alignment and mean matching: excessive emphasis on preserving covariance structures may inadvertently weaken information on the inherent properties of each class.

Effect of λ_1 and λ_2. The regularization weights λ_1 (for variance minimization) and λ_2 (for off-diagonal covariance maximization) address distinct aspects of feature distribution. In the pre-trained model, last-layer features exhibit high concentration within classes while carrying high-level semantic information. λ_1 explicitly constrains feature diversity across distilled samples, whereas λ_2 encourages dimension decorrelation to enhance feature representational capacity. As shown in Figure 3b and 3c for CIFAR-100 with IPC=50, optimal performance is achieved with $\lambda_1 = 0.2$ and $\lambda_2 = 4$. Notably, large λ_1 values may degrade performance due to excessive expansion of the feature space, which may compromise class discriminability. In contrast, performance remains relatively stable across variations in λ_2, indicating lower sensitivity to this parameter.

5.5. Visualization

To assess whether our method preserves the feature diversity of the original dataset, we present visual comparisons between the distilled data from SRe2L and our DEDA on CIFAR-100 and ImageNet-1K in Figure 4. Our empirical observations yield two critical insights: (1) On low-resolution data like CIFAR-100, our DEDA demonstrates superior class discriminability compared to SRe2L, with distilled samples effectively preserving class-specific semantic information. This enhanced performance validates the effectiveness of our class-specific Gaussian distribution matching strategy in maintaining inter-class distinctions. (2) On high-resolution data like ImageNet-1K, our DEDA not only maintains class discriminability but also captures richer semantic diversity. For instance, the distilled images of the 'sorrel' class (red-brown horse) preserve various fine-grained attributes, including equine postures, anatomical features, and color variations. This underscores the robustness of our method across different data scales.

6. Conclusion

In this paper, we address two key limitations in modern dataset distillation: (1) the feature alignment of distilled data to original data overlooks crucial inter-feature correlations, and (2) the compact feature space of the pre-trained model's last layer leads to a lack of diversity in the distilled data. To overcome these challenges, we propose a novel Diversity-Enhanced Distribution Alignment (DEDA) method for dataset distillation. DEDA enhances diversity by aligning the distilled data with the Gaussian distribution of the original data and implementing a diversity-enriched regularizer at the last layer. Our experimental evaluation across multiple benchmarks demonstrates that DEDA consistently outperforms existing methods while maintaining computational efficiency. Although DEDA shows promising results on large-scale datasets, its performance declines as distilled samples increase (e.g., IPC=100), highlighting the need for further exploration to balance class-specific information retention and diversity enhancement.

7. Acknowledgments

This work was supported by the Chinese Academy of Sciences under grant No. XDB0690302.

References

[1] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In *ICLR*, 2022. 5

[2] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicregl: Self-supervised learning of local visual features. *Advances in Neural Information Processing Systems*, 35:8799–8810, 2022. 5

[3] 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 (CVPR)*, pages 10718–10727, 2022. 2, 3, 6

[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. 2

[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. 6

[6] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17057–17066, 2024. 3

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1

[8] 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

[9] Jiawei Du, Qin Shi, and Joey Tianyi Zhou. Sequential subset matching for dataset distillation. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 2

[10] Jiawei Du, Xin Zhang, Juncheng Hu, Wenxin Huang, and Joey Tianyi Zhou. Diversity-driven synthesis: Enhancing dataset distillation through directed weight adjustment. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2024. 3, 5, 6

[11] Zongxion Geng, Jiahui andg Chen, Yuandou Wang, Herbert Woisetschlaeger, Sonja Schimmler, Ruben Mayer, Zhiming Zhao, and Chunming Rong. A survey on dataset distillation: Approaches, applications and future directions. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2023. 1

[12] Jianyang Gu, Saeed Vahidian, Vyacheslav Kungurtsev, Haonan Wang, Wei Jiang, Yang You, and Yiran Chen. Efficient dataset distillation via minimax diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15793–15803, 2024. 3

[13] Zenghao Guan, Yucan Zhou, and Xiaoyan Gu. Capture global feature statistics for one-shot federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 16942–16950, 2025. 1

[14] Ziyao Guo, Kai Wang, George Cazenavette, Hui Li, Kaipeng Zhang, and Yang You. Towards lossless dataset distillation via difficulty-aligned trajectory matching. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024. 2

[15] 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. 6, 7

[16] 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

[17] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 6

[18] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International journal of computer vision*, 128(7):1956–1981, 2020. 1

[19] Yann Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015. 6

[20] Saehyung Lee, Sanghyuk Chun, Sangwon Jung, Sangdoo Yun, and Sungroh Yoon. Dataset condensation with contrastive signals. In *International Conference on Machine Learning*, pages 12352–12364. PMLR, 2022. 2

[21] Shiye Lei and Dacheng Tao. A comprehensive survey of dataset distillation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1

[22] Hongcheng Li, Yucan Zhou, Xiaoyan Gu, Bo Li, and Weiping Wang. Diversified semantic distribution matching for dataset distillation. In *Proceedings of the ACM International Conference on Multimedia (MM)*, 2024. 3

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 1

[24] Dai Liu, Jindong Gu, Hu Cao, Carsten Trinitis, and Martin Schulz. Dataset distillation by automatic training trajectories. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024. 2

[25] Ping Liu and Jiawei Du. The evolution of dataset distillation: Toward scalable and generalizable solutions. *arXiv preprint arXiv:2502.05673*, 2025. 1

[26] Yanqing Liu, Jianyang Gu, Kai Wang, Zheng Zhu, Wei Jiang, and Yang You. Dream: Efficient dataset distillation by representative matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17314–17324, 2023. 2

[27] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*, pages 116–131, 2018. 7

[28] Timothy Nguyen, Roman Novak, Lechao Xiao, and Jaehoon Lee. Dataset distillation with infinitely wide convolutional networks. *Advances in Neural Information Processing Systems*, 34:5186–5198, 2021. 6

[29] Vardan Papyan, XY Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020. 5

[30] Mohammad Pezeshki, Oumar Kaba, Yoshua Bengio, Aaron C Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. *Advances in Neural Information Processing Systems*, 34:1256–1272, 2021. 5

[31] Ahmad Sajedi, Samir Khaki, Ehsan Amjadian, Lucy Z Liu, Yuri A Lawryshyn, and Konstantinos N Plataniotis. Datadam: Efficient dataset distillation with attention matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17097–17107, 2023. 3, 6

[32] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 7

[33] 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 (CVPR)*, pages 16709–16718, 2024. 3, 6

[34] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. 2015. 7

[35] Duo Su, Junjie Hou, Weizhi Gao, Yingjie Tian, and Bowen Tang. D^4m: Dataset distillation via disentangled diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5809–5818, 2024. 3

[36] 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 (CVPR)*, 2024. 3

[37] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019. 7

[38] Zixin Tang, Haihui Fan, Xiaoyan Gu, Yang Li, Bo Li, and Xin Wang. ELSEIR: A privacy-preserving large-scale image retrieval framework for outsourced data sharing. In *Proceedings of the 2024 International Conference on Multimedia Retrieval, ICMR 2024*. ACM, 2024. 1

[39] 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 (CVPR)*, pages 12196–12205, 2022. 3, 6

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

[41] Lingao Xiao and Yang He. Are large-scale soft labels necessary for large-scale dataset distillation? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 2, 3, 5, 6

[42] Zeyuan Yin and Zhiqiang Shen. Dataset distillation via curriculum data synthesis in large data era. *Transactions on Machine Learning Research*, 2024. 1, 3, 4

[43] Zeyuan Yin, Eric Xing, and Zhiqiang Shen. Squeeze, recover and relabel: Dataset condensation at imagenet scale from a new perspective. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 2, 3, 4, 5, 6, 1

[44] Ruonan Yu, Songhua Liu, and Xinchao Wang. A comprehensive survey to dataset distillation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(1):150–170, 2023. 1

[45] Hansong Zhang, Shikun Li, Fanzhao Lin, Weiping Wang, Zhenxing Qian, and Shiming Ge. DANCE: Dual-view distribution alignment for dataset condensation. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2024. 3

[46] Hansong Zhang, Shikun Li, Pengju Wang, Dan Zeng, and Shiming Ge. M3d: Dataset condensation by minimizing maximum mean discrepancy. In *The 38th Annual AAAI Conference on Artificial Intelligence (AAAI)*, 2024. 3

[47] Bo Zhao and Hakan Bilen. Dataset condensation with gradient matching. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 2, 3, 6

[48] Bo Zhao and Hakan Bilen. Dataset condensation with differentiable siamese augmentation. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 12674–12685, 2021. 2

[49] 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. 3, 6

[50] 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. 3