Generalized Large-Scale Data Condensation via Various Backbone and Statistical Matching

Shitong Shao[1,3]   Zeyuan Yin[1]   Muxin Zhou[1]   Xindong Zhang[2,3]   Zhiqiang Shen[1,*]

[1]Mohamed bin Zayed University of AI [2]Hong Kong Polytechnic University [3]OPPO Research

1090784053sst@gmail.com, {zeyuan.yin,muxin.zhou,zhiqiang.shen}@mbzuai.ac.ae
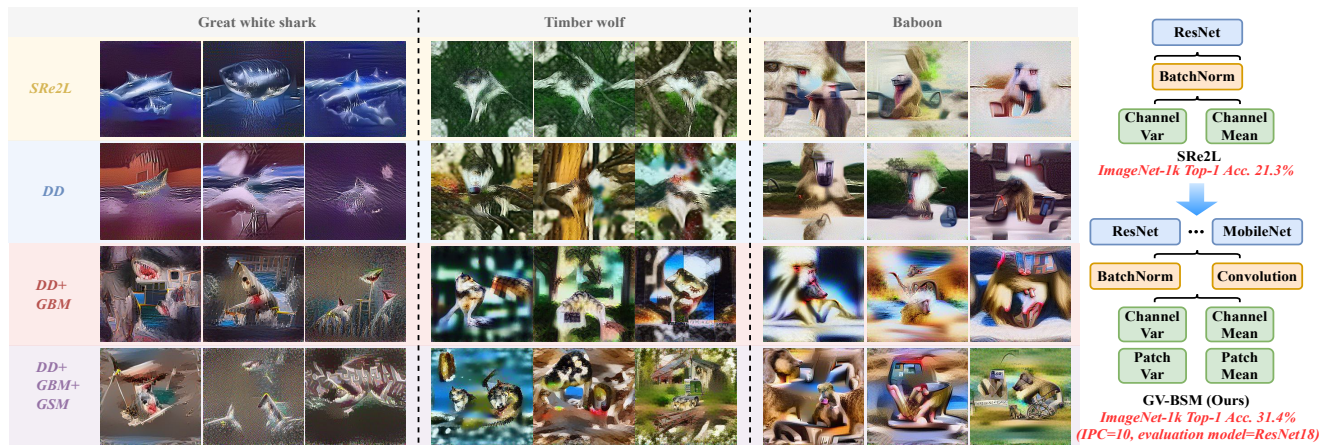
17901410r@connect.polyu.hk, *:Corresponding author

Figure 1. **Left:** Our proposed **G-VBSM** consists of three novel and effective modules, named DD, GSM and GBM. The richness and quality of information in the synthetic data have been significantly enhanced compared with the *baseline* SRe2L through the sequential merging of DD, GBM, and GSM. **Right:** G-VBSM prioritizes "generalized matching" to ensure consistency between distilled and complete datasets across various backbones, layers, and statistics, and achieves the highest accuracy 31.4% on ImageNet-1k under IPC 10.

Abstract

The lightweight "local-match-global" matching introduced by SRe2L successfully creates a distilled dataset with comprehensive information on the full 224×224 ImageNet-1k. However, this one-sided approach is limited to a particular backbone, layer, and statistics, which limits the improvement of the generalization of a distilled dataset. We suggest that sufficient and various "local-match-global" matching are more precise and effective than a single one and have the ability to create a distilled dataset with richer information and better generalization ability. We call this perspective "generalized matching" and propose Generalized Various Backbone and Statistical Matching (G-VBSM) in this work, which aims to create a synthetic dataset with densities, ensuring consistency with the complete dataset across various backbones, layers, and statistics. As experimentally demonstrated, G-VBSM is the first algorithm to obtain strong performance across both small-scale and large-scale datasets. Specifically, G-VBSM achieves performances of 38.7% on CIFAR-100, 47.6% on Tiny-ImageNet, and 31.4% on the full 224×224 ImageNet-1k, respectively[1]. These results surpass all SOTA methods by margins of 3.9%, 6.5%, and 10.1%, respectively.

1. Introduction

With the development of deep learning, the number of model parameters and the quantity of training data have become increasingly large [4, 37]. Researchers have tried to minimize the training overhead while preventing a decline in the generalization ability. Data condensation (DC), also known as Dataset distillation, first introduced by Wang *et al.* [32], aims to alleviate the training burden by synthesizing a small yet informative distilled dataset derived from the complete training dataset, while ensuring that the behav-

---

[1]Settings: CIFAR-100 with 128-width ConvNet under 10 images per class (IPC), Tiny-ImageNet with ResNet18 under 50 IPC, and ImageNet-1k with ResNet18 under 10 IPC.

ior of the distilled dataset on the target task remains consistent with that of the complete dataset. The extremely compressed distilled dataset contains sufficiently valuable information and have the potential for fast model training, and have been become a popular choice for different downstream application, like federated learning [7, 22], continual learning [14, 22, 40], neural architecture search [24, 41, 42] and 3D point clouds [30].

A persistent problem that researchers have been working to solve [2, 9, 12, 42] in DC is the substantial data synthesis overhead [19, 36], which hinders its application in real-world large-scale datasets (*e.g.*, ImageNet-1k) training. Typical performance matching [15, 32, 44] and trajectory matching [1, 3] unroll recursive computation graphs, requiring substantial GPU memory and resulting in prohibitive training costs. Zhao *et al.* [42] proposed gradient matching to address this, synthesizing distilled datasets by matching gradients from synthetic and real data in a single step. However, gradient computation and matching remain time-consuming [9], leading to the proposal of distribution matching [41]. This method and its variants [31] employ a network-based feature extractor to embed both synthetic and real data into a high-dimensional Hilbert space, then perform distribution matching. The training load for this direct, single-step process stems only from one gradient update of the synthetic data and the feature extractor [36]. Unfortunately, all of the above mentioned improved methods still have extremely large training overheads on the full 224×224 ImageNet-1k.

Recently, SRe2L [34] accomplished data condensation for the first time on the full 224×224 ImageNet-1k [18], achieving Top-1 validation accuracy 21.3% with ResNet18 under IPC 10. This method outperformed the latest state-of-the-art TESLA [2], which conducted on a low-resolution version of ImageNet-1k, by being 16× faster and improved performance by a margin of 13.6%. SRe2L is inspired by DeepInversion [33] and aims to match statistics in Batch-Norm generated from synthetic and real data. We reevaluate the success of SRe2L through the lightweight "local-match-global" essentially. The "local-match-global" refers to utilizing more comprehensive information (*e.g.*, statistics in BatchNorm), generated from the model using the complete dataset (global), to guide the parameter update of the distilled dataset (local).

However, such lightweight and effective matching in SRe2L is singular, depending on the particular layer (*i.e.*, BatchNorm), model (*i.e.*, ResNet18), and statistics (*i.e.*, channel mean/variance). Intuitively, sufficient "local-match-global" matching can result in more accurate and rational supervision than a single one, further enhancing the generalization of the distilled dataset. We call this perspective "generalized matching" and suggest that the distilled dataset is likely to perform consistent with the complete

dataset on the evaluation model, provided that there is sufficient variety in backbones, layers, and statistics used for matching.

Inspired by this, we propose **G**eneralized **V**arious **B**ackbone and **S**tatistical **M**atching (**G-VBSM**) to fulfill "generalized matching". G-VBSM comprises three integral and effective parts named data densification (DD), generalized statistical matching (GSM), and generalized backbone matching (GBM). DD is utilized to ensure that the images within each class are linearly independent, thereby enhancing the (intra-class) diversity of the distilled dataset. This ultimately guarantees that "generalized matching" preserves the rich and diverse information within the synthetic data. GBM and GSM are designed to implement "generalized matching". Among them, GSM traverses the complete dataset without computing and updating the gradient, to record the statistics of Convolution at both the patch and channel levels. These statistics are subsequently utilized for matching during the data synthesis phase, in conjunction with the channel-level statistics in BatchNorm. Furthermore, GBM aims to ensure consistency between distilled and complete datasets across various backbones, enhancing matching sufficiency and leading to strong generalization in the evaluation phase. In particular, G-VBSM also ensures the efficiency of dataset condensation through a series of strategies, as mentioned in Sec. 3.

Extensive experiments on CIFAR-10, CIFAR-100, Tiny-ImageNet, and the full 224×224 ImageNet-1k, demonstrating that G-VBSM is the first algorithm that performs well on both small-scale and large-scale datasets. Specifically, we not only verify through ablation studies that GSM, GBM and DD are consistently reliable, but also achieve the highest 38.7%, 47.6% and 31.4% on CIFAR-100 (128-width ConvNet), Tiny-ImageNet (ResNet18), and the full 224×224 ImageNet-1k (ResNet18) under images per class (IPC) 10, 50 and 10, respectively, which outperforms all previous state-of-the-art (SOTA) methods by 3.9%, 6.5% and 10.1%, respectively.

2. Background

Dataset condensation (DC) represents a data synthesis procedure that aims to compress a complete, large dataset $\mathcal{T} = \{(X_i, y_i)\}_{i=1}^{|\mathcal{T}|}$ into a smaller, distilled dataset $\mathcal{S} = \{(\tilde{X}_i, \tilde{y}_i)\}_{i=1}^{|\mathcal{S}|}$, subject to $|\mathcal{S}| \ll |\mathcal{T}|$, while ensuring that an arbitrary evaluation model $f_{\text{eval}}(\cdot)$ trained on $\mathcal{S}$ yields similar performance to one trained on $\mathcal{T}$. Classical data distillation algorithms invariably require the candidate model $f_{\text{cand}}(\cdot)$ to execute one or more steps on $\mathcal{S}$ to update its parameter θ_{cand}, subsequently achieving matching in terms of performance [15, 32], gradient [42], trajectory [1, 2], or distribution [31, 41]. The process $\theta_{\text{cand}} - \alpha \nabla_{\theta_{\text{cand}}} \ell(f_{\text{cand}}(\tilde{X}), \tilde{y})$, where $\ell(\cdot, \cdot)$ and $(\tilde{X}, \tilde{y})$ represent the loss function and
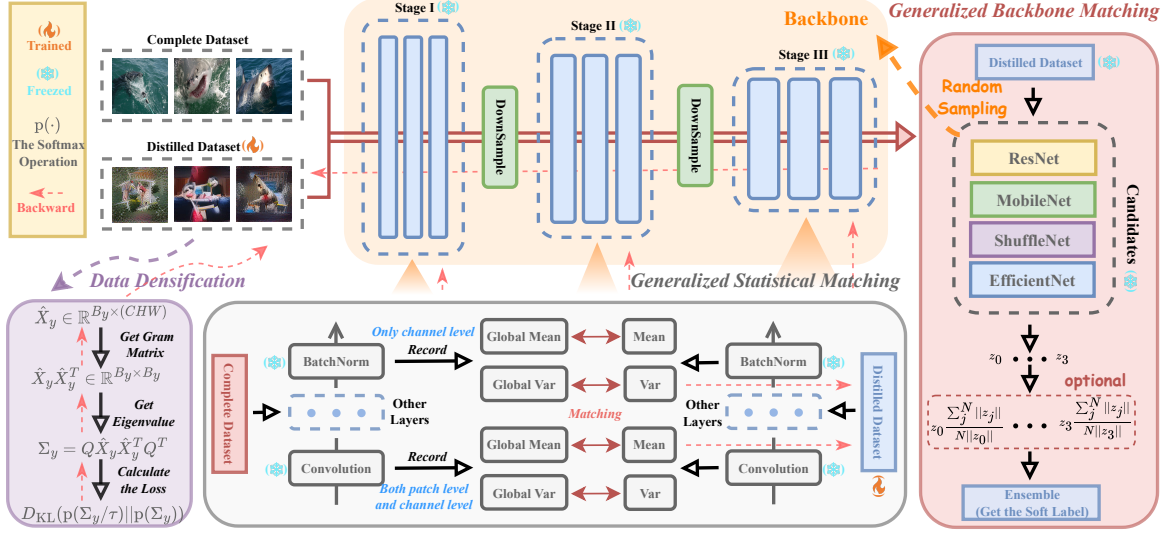
Figure 2. The overview of G-VBSM on the full 224×224 ImageNet-1k, which ensures the consistency between the distilled and the complete datasets across various backbones, layers and statistics via "generalized matching".

a batch sampled from $\mathcal{S}$, respectively, is notably time-consuming. Consequently, even the relatively swiftest distribution matching [20, 31] is slower than the recent proposed SRe2L [34]. In fact, SRe2L is the only workable way to achieve DC on the full 224×224 ImageNet-1k, as it requires updating the parameters of the synthetic data only once per iteration.

SRe2L [34] encompasses three incremental subprocesses: Squeeze, Recover, and Relabel. Squeeze is designed to train $f_{\text{cand}}(\cdot)$ containing BatchNorm in a standard manner, aiming to record the global channel mean $\mathbf{BN}_l^{\text{CM}}$ and channel variance $\mathbf{BN}_l^{\text{CV}}$ (l refers to the index of the l-th layer) via exponential moving average (EMA), extracted from $\mathcal{T}$, for subsequent matching in Recover. In Recover after that, given the channel mean $\mu_l(\tilde{X})$ and channel variance $\sigma_l^2(\tilde{X})$ in l-th BatchNorm obtained from $\mathcal{S}$, the statistical matching loss function can be formulated as

$$\mathcal{L}_{\text{BN}}(\tilde{X}) = \sum_l \left\| \mu_l(\tilde{X}) - \mathbf{BN}_l^{\text{CM}} \right\|_2 + \left\| \sigma_l^2(\tilde{X}) - \mathbf{BN}_l^{\text{CV}} \right\|_2. \quad (1)$$

Based on this, we can give the entire optimization objective in Recover as

$$\arg\min_{\tilde{X}} \mathcal{L}_{\text{BN}}(\tilde{X}) + \ell(f_{\text{cand}}(\tilde{X}), y), \quad (2)$$

where y denotes the ground truth label. Moreover, SRe2L assigns soft labels $\tilde{y}$ to the synthetic data, utilizing the logit-based distillation [6, 8] to improve the generalization ability of the distilled dataset. This can be denoted as

$$\tilde{y} = \text{softmax}(f_{\text{cand}}(\tilde{X})/\tau), \quad (3)$$

where τ denotes the temperature to regulate the smoothness of the soft labels, thereby enhancing the distilled dataset's potential for generalization to unseen evaluation

models. The generated soft label can be stored on disk using FKD [23] so as not to defeat the purpose of DC. A crucial point of SRe2L is that BatchNorm calculates the statistics of the entire dataset using EMA, thereby providing a comprehensive and representative matching for the distilled dataset. Encouraged by this, our research focuses on applying sufficient 'local-match-global" matching to achieve "generalized matching".

3. Method

The comprehensive and detailed framework of our proposed **G**eneralized **V**arious **B**ackbone and **S**tatistical **M**atching (**G-VBSM**) is illustrated in Fig. 2. In essence, G-VBSM employs the lightweight regularization strategy data densification (DD) to ensure both the diversity and density of the distilled dataset, ensuring that the potential of "generalized matching" can be fully exploited. Moreover, generalized backbone matching (GBM) and generalized statistical matching (GSM) are utilized to achieve "generalized matching" by performing "local-match-global" matching across various backbones, layers, and statistics. In particular, the efficiency and effectiveness of DD, GBM, GSM, SRe2L, and TESLA are illustrated in Fig. 5.

3.1. Data Densification

As illustrated in Fig. 3, the synthetic data generated by SRe2L exhibit excessive similarity within the same class, leading to a lack of diversity. Consequently, even if "generalized matching" preserves sufficient valuable information within a single image, the aggregate information content across all images within the same class does not increase effectively, which ultimately prevents "generalized match-

Figure 3. Visualization of the mean cosine similarity between pairwise samples within the same class on ImageNet-1k under IPC 10.

Figure 4. The illustration of the original loop and the reorder loop.

ing" from being sufficiently advantageous. Data densification (DD) is proposed to address this by ensuring the data $\tilde{X}$ has full rank in the batch dimension, thereby guaranteeing that samples in each class are linearly independent, and ultimately ensuring that the data is diverse and abundant to fully exploit the potential of "generalized matching".

To execute this pipeline, $\tilde{X}$ first needs to be downsampled to reduce the computational cost of eigenvalue decomposition:

$$\hat{X} = \text{AvgPool2d}(\tilde{X}), \ s.t. \ \hat{X} \in \mathbb{R}^{B \times C \times 32 \times 32}, \quad (4)$$

where B and C represent the batch size and the number of channels, respectively. Afterward, we reshape $\hat{X}$ from $B \times C \times 32 \times 32$ to $B \times (1024C)$ and perform matrix multiplication in each class y to obtain the set of the Gram matrix $\{\hat{X}_y \hat{X}_y^T\}_{y \in \mathcal{Y}}$, where $\mathcal{Y}$ refers to a set of all classes existing in this batch. And $\hat{X}_y$ is a subbatch with class y. Note that the alternative form $\{\hat{X}_y^T \hat{X}_y\}_{y \in \mathcal{Y}}$ is not desirable, as it is applicable only for dimensionality reduction in feature dimensions, which is why we do not consider singular value decomposition (SVD). To render $\hat{X}_y \hat{X}_y^T$ as full-rank as possible, we introduce the data densification loss in Eq. 5.

$$\mathcal{L}_{\text{DD}}(\tilde{X}) = \sum_{y \in \mathcal{Y}} D_{\text{KL}}(\text{stop\_grad}(p(\Sigma_y / \tau)) || p(\Sigma_y)), \quad (5)$$

where Σ_y, τ, stop_grad$(\cdot)$ and $p(\cdot)$ refer to the eigenvalues of $\hat{X}_y \hat{X}_y^T$, the temperature, the stop gradient operator and the softmax function, respectively. And $D_{\text{KL}}(\cdot || \cdot)$ denotes Kullback-Leibler divergence. As demonstrated in Sec. E, the diversity of the data is significantly enhanced by the employment of Eq. 5. In our experiment, τ is set as 4 in default and we do not assign a deliberate weight (set to 1 by default) to $\mathcal{L}_{\text{DD}}$ because $\mathcal{L}_{\text{DD}} \equiv 0$ at the early 10% of the iterations. In other words, DD is quite stable and the optimization objective $\arg\min_{\{\Sigma_y\}_{y \in \mathcal{Y}}} \mathcal{L}_{\text{DD}}$ is equivalent to $\arg\min_{\{\Sigma_y\}_{y \in \mathcal{Y}}} \sum_y \sigma^2(\Sigma_y)$.

Technical Detail. A problem that warrants attention is that in the SRe2L's implementation [34], having merely a single sample in each class of a batch indicates insufficient
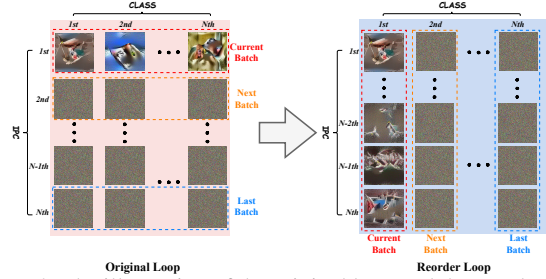
to execute DD under the order of the original loop, as depicted in Fig. 4 (Left). A simple solution is to translate the original loop to the reorder loop, as shown in Fig. 4 (Right). However, our experiment on ResNet50 (*i.e.*, the evaluation model) substantiates that this solution suffers a 2.6% accuracy degradation (details can be found in Sec. 4.1) on ImageNet-1k under IPC 10. The reason is that the number of classes in each iteration within the reorder loop is insufficient, preventing a batch have the ability to match the global statistics in BatchNorm (*i.e.*, $\textbf{BN}_l^{\text{CM}}$ and $\textbf{BN}_l^{\text{CV}}$). Motivated by score distillation sampling (SDS) [17], we update the statistics during data synthesis via EMA to solve this issue, so that the statistics of all past batches can assist the statistics of the current batch match $\textbf{BN}_l^{\text{CM}}$ and $\textbf{BN}_l^{\text{CV}}$:

$$\mu_l^{\text{total}} = \alpha \mu_l^{\text{total}} + (1 - \alpha)\mu_l(\tilde{X}), \sigma_l^{2,\text{total}} = \alpha\sigma_l^{2,\text{total}} + (1 - \alpha)\sigma_l^2(\tilde{X}),$$

$$\mathcal{L}'_{\text{BN}}(\tilde{X}) = \sum_l ||\mu_l(\tilde{X}) - \textbf{BN}_l^{\text{CM}} - \text{stop\_grad}(\mu_l(\tilde{X}) - \mu_l^{\text{total}})||_2$$

$$+ ||\sigma_l^2(\tilde{X}) - \textbf{BN}_l^{\text{CV}} - \text{stop\_grad}(\sigma_l^2(\tilde{X}) - \sigma_l^{2,\text{total}})||_2. \quad (6)$$

The derivation of $\mathcal{L}'_{\text{BN}}(\tilde{X})$ can be found in Appendix B. We call this lightweight strategy as "match in the form of score distillation sampling" and have demonstrated its effectiveness and feasibility in our ablation studies.

3.2. Generalized Backbone Matching

Performing data synthesis only on a single pre-trained model is not able to enjoy ensemble gains from various backbones. Meanwhile, classical DC algorithms such as MTT [1] and FTD [3] obtain performance improvements from multiple randomly initialized backbones. Therefore, introducing generalized backbone matching (GBM) to apply various backbones for data synthesis is a desirable choice for "generalized matching". It ensures a number of "local-match-global" matching nearly $N_\text{b}\times$ higher compared to just depending on a single backbone, where N_b denotes the number of backbones. Regrettably, unrolling various backbone computational graphs in parallel for data synthesis is extremely expensive at the computational cost level. A solution is to randomly sample a backbone from the candidates in per iteration. This simple yet effective strategy not only ensures computational efficiency but also allows the data synthesis process to benefit from the diverse and multifaceted matching that the various backbones pro-

vide. We denote this pipeline as

$$f_{\text{cand}}(\cdot) \sim \mathcal{U}(\mathbb{S}), \ \mathbb{S} = \{\text{ResNet18}, \cdots, \text{ShuffleNetV2-0.5}\}. \quad (7)$$

To maintain backbone consistency in both the data synthesis and soft label generation phases, we introduce a backbone ensemble during soft label generation:

$$\tilde{z} = \begin{cases} \displaystyle\sum_{f_{\text{cand}} \in \mathbb{S}} \frac{\sum_{g \in \mathbb{S}} ||g(\tilde{X})||_{\text{F}}}{|\mathbb{S}| ||f_{\text{cand}}(\tilde{X})||_{\text{F}}} \frac{f_{\text{cand}}(\tilde{X})}{|\mathbb{S}|}, & \text{w/ LN}, \\ \displaystyle\sum_{f_{\text{cand}} \in \mathbb{S}} \frac{f_{\text{cand}}(\tilde{X})}{|\mathbb{S}|}, & \text{w/o LN}, \end{cases} \quad (8)$$

where $|| \cdot ||_{\text{F}}$ and LN refer to the Frobenius norm and Logit Normalization, respectively. Thus, we can obtain the soft label by $\tilde{y} = \text{softmax}(\tilde{z}/\tau)$. Particularly, the use of LN is optional, as demonstrated by our ablation studies; it is beneficial for ResNet18 but not for ResNet50 or ResNet101. Moreover, it's important to highlight that we apply a parallel mechanism for soft label generation since it provides a significantly lower computational cost, less than $1/30$ of that required for data synthesis, thus making the computational overhead negligible.

3.3. Generalized Statistical Matching

Only ensuring backbone diversity is insufficient to fully exploit the potential of "generalized matching". In this subsection, we aim to introduce additional statistics for matching during the data synthesis phase. Prior methods [33, 34] only utilize BatchNorm since the presence of global information statistics within the pre-trained model, with no apparent solution ready for other layers (*e.g.*, Convolution). Retraining the model is the simplest way to address this problem, followed by updating the other layers' parameters through EMA. This approach is obviously impractical, as it defeats the purpose of using gradient descent. By contrast, we propose to allow the pre-trained model $f_{\text{eval}}(\cdot)$ to loop through the training dataset $\mathcal{T}$ once without calculating the gradient to obtain the global statistics of Convolution, thus serving "local-match-global":

$$\mathbf{Conv}_l^{\mathbf{CM}} = \frac{1}{|\mathcal{T}|} \sum_i^{|\mathcal{T}|} \mathbf{CM}_i^l, \ \mathbf{Conv}_l^{\mathbf{CV}} = \frac{1}{|\mathcal{T}|} \sum_i^{|\mathcal{T}|} \mathbf{CV}_i^l,$$

$$\mathbf{Conv}_l^{\mathbf{PM}} = \frac{1}{|\mathcal{T}|} \sum_i^{|\mathcal{T}|} \mathbf{PM}_i^l, \ \mathbf{Conv}_l^{\mathbf{PV}} = \frac{1}{|\mathcal{T}|} \sum_i^{|\mathcal{T}|} \mathbf{PV}_i^l. \quad (9)$$

Here, $\mathbf{CM}_i^l \in \mathbb{R}^{C_l}$, $\mathbf{CV}_i^l \in \mathbb{R}^{C_l}$, $\mathbf{PM}_i^l \in \mathbb{R}^{\lceil \frac{H}{N_l^p} \rceil \times \lceil \frac{W}{N_l^p} \rceil}$ and $\mathbf{PV}_i^l \in \mathbb{R}^{\lceil \frac{H}{N_l^p} \rceil \times \lceil \frac{W}{N_l^p} \rceil}$ refer to the channel mean, the channel variance, the patch mean and the patch variance, respectively, for the l-th Convolution when the input to $f_{\text{cand}}(\cdot)$ is the i-th batch, where C_l and $\lceil \frac{H}{N_l^p} \rceil \times \lceil \frac{W}{N_l^p} \rceil$ denote the number of channels and patches of the l-th Convolution, respectively. We define N_l^p as 4, 4, 4 and 16 by default

on CIFAR-10, CIFAR-100, Tiny-ImageNet and ImageNet-1k, respectively. After obtaining the global channel mean $\mathbf{Conv}_l^{\mathbf{CM}}$, the global channel variance $\mathbf{Conv}_l^{\mathbf{CV}}$, the global patch mean $\mathbf{Conv}_l^{\mathbf{PM}}$ and the global patch variance $\mathbf{Conv}_l^{\mathbf{PV}}$, we can store them in a disk thus avoiding secondary calculations. In the data synthesis phase, we introduce $\mathcal{L}'_{\text{Conv}}(\tilde{X})$ in Eq. 10 to accomplish joint matching with $\mathcal{L}'_{\text{BN}}(\tilde{X})$.

$$\mathcal{L}'_{\text{Conv}}(\tilde{X}) = \sum_l ||\mu_l^c(\tilde{X}) - \mathbf{Conv}_l^{\mathbf{CM}} - \text{stop\_grad}(\mu_l^c(\tilde{X}) - \mu_l^{c,\text{total}})||_2$$

$$+ ||\sigma_l^{c,2}(\tilde{X}) - \mathbf{Conv}_l^{\mathbf{CV}} - \text{stop\_grad}(\sigma_l^{c,2}(\tilde{X}) - \sigma_l^{c,2,\text{total}})||_2,$$

$$+ ||\mu_l^p(\tilde{X}) - \mathbf{Conv}_l^{\mathbf{PM}} - \text{stop\_grad}(\mu_l^p(\tilde{X}) - \mu_l^{p,\text{total}})||_2$$

$$+ ||\sigma_l^{p,2}(\tilde{X}) - \mathbf{Conv}_l^{\mathbf{PV}} - \text{stop\_grad}(\sigma_l^{p,2}(\tilde{X}) - \sigma_l^{p,2,\text{total}})||_2, \quad (10)$$

where $\mu_l^{c,\text{total}}$, $\sigma_l^{c,2,\text{total}}$, $\mu_l^{p,\text{total}}$, and $\sigma_l^{p,2,\text{total}}$ are each updated from the channel mean $\mu_l^c(\tilde{X})$, channel variance $\sigma_l^{c,2}(\tilde{X})$, patch mean $\mu_l^p(\tilde{X})$, and patch variance $\sigma_l^{p,2}(\tilde{X})$ respectively, all obtained via EMA from the current batch. In experiments, we discovered that Eq. 10 causes a sightly larger computational burden, so we randomly drop the matching of statistics with a probability of β_{dr} to ensure the efficiency of GSM.

Loss Function in the Evaluation Phase. Unlike DD, GBM and GSM are designed to create a distilled dataset that is enriched with information. Here, we introduce an enhancement to the loss function tailored specifically for the evaluation phase. Essentially, the evaluation phase is a knowledge distillation framework for transferring knowledge from a pre-trained model to the evaluation model. SRe2L utilizes $D_{\text{KL}}(\tilde{y}||\text{softmax}(f_{\text{eval}}(\tilde{X})/\tau))$ as the loss function and experimentally illustrates that it improves performance by roughly 10%. As established in SRe2L, an increase in temperature τ correlates with enhanced performance of the evaluation model. Inspired by this and $\tau^2 D_{\text{KL}}(\text{softmax}(p)/\tau||\text{softmax}(q)/\tau)$ is equivalent to $\frac{1}{2C}||p-q||_2^2$ when $\tau \to +\infty$ [10], we introduce a novel loss function MSE+$\gamma \times$GT to avoid numerical error caused by the large τ and improve the generalization of the distilled dataset without any additional overhead (ignore the weights τ^2 and $\frac{1}{2C}$):

$$\mathcal{L}_{\text{eval}}(\tilde{X}, \tilde{y}, y) = ||f_{\text{eval}}(\tilde{X}) - \tilde{z}||_2^2 - \gamma \mathbf{y} \log(\text{softmax}(f_{\text{eval}}(\tilde{X}))), \quad (11)$$

where y represents the one-hot encoding (*w.r.t.* the ground truth label y). As illustrated in Fig. 5, simply replacing the loss function with MSE+0.1$\times$GT (γ is set as 0.1) in SRe2L improves the performance of ResNet18 (*i.e.*, the evaluation model) by a margin of 0.9% on ImageNet-1k under IPC 10.

4. Experiment

We perform comparison experiments on the large-scale dataset including the full 224×224 ImageNet-1k [18] and the small-scale datasets including Tiny-ImageNet [26],
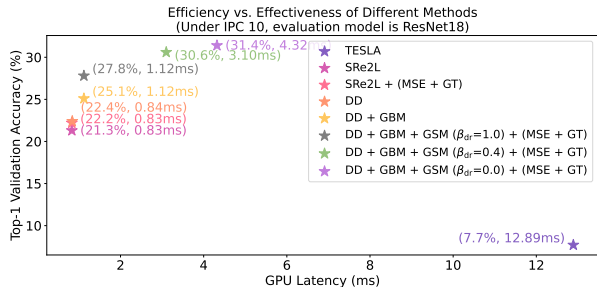
Figure 5. Comparison of the effectiveness and efficiency of G-VBSM components. Among them, "DD+GBM+GSM (β_{dr}=0.0)+(MSE+GT)" represents the comprehensive G-VBSM.

CIFAR-10/100 [11]. To highlight that our proposed G-VBSM is designed for large-scale datasets, all ablation experiments are performed on ImageNet-1k.

Hyperparameter Settings. We prioritize selecting various convolution-based backbone, ensuring the maximal difference in architecture, while also adhering to the criterion of minimizing the number of parameters, an approach empirically demonstrated to be superior [34, 38]. On ImageNet-1k, we skip the model pre-training phase by directly using Torchvision's open source pre-training weights [16] of {ResNet18 [5], MobileNetV2 [21], EfficientNet-B0 [25], ShuffleNetV2-0.5 [39]}. For the remaining dataset, we all train {128-width ConvNet [39], WRN-16-2 [37], ResNet18 [5], ShuffleNetV2-0.5 [39], MobileNetV2-0.5 [21]} from scratch with few epoch in the model pre-training phases. Gray cells in all tables represent the highest performance. Meanwhile, (R18), (R50), etc. in all tables, represent the evaluation models. More details about the remaining method hyperparameter settings and additional ablation studies can be found in Appendix A and Appendix C & D, respectively.

4.1. Ablation Studies

The Trade-off between Efficiency and Effectiveness. The performance improvements and increased GPU latency for components of G-VBSM in comparison to the *baseline* SRe2L, are illustrated in Fig. 5. Clearly, DD, GBM, and GSM are highly effective in enhancing the generalization of distilled datasets. Compared with GSM, both DD and GBM are slightly lightweight and efficient. With DD and GBM alone – *i.e.*, DD+GBM+GSM (β_{dr}=1.0)+(MSE+GT) – the accuracy of this approach surpasses that of SRe2L by a significant margin of 6.5%. Additionally, the comprehensive G-VBSM, DD+GBM+GSM (β_{dr}=0.0)+(MSE+GT), further enhances the performance of DD+GBM+GSM (β_{dr}=1.0)+(MSE+GT) by a notable margin of 3.6%. Note that MSE+$\gamma \times$GT (γ is set as 0.1) is also extremely critical, for SRe2L and DD+GBM boosting 0.9% and 2.7%, respectively.

α\Evaluation Model	ResNet18 (MSE+0.1×GT)	ResNet50 (MSE+0.1×GT)
0.0	30.4%	31.9%
0.4	31.4%	34.5%
0.8	31.9%	36.4%

Table 1. ImageNet-1k Top-1 Acc. about different α under IPC 10.

Matching in the Form of SDS. Benefiting from this matching form, G-VBSM can achieve the best performance with small batch sizes (*e.g.*, 40 in ImageNet-1k and 50 in CIFAR-100) in different settings, as demonstrated in our comparative experiments. Table 1 demonstrates the impact of the factor α for performing the EMA update on the final performance achieved by G-VBSM, where α=0 indicates that $\mathcal{L}'_{BN}$ degenerates to $\mathcal{L}_{BN}$. We can conclude that the SDS matching form is feasible and the generalization of the distilled dataset improves with increasing α.

Method	Evaluation Model			
	DeiT-Tiny	ResNet18	MobileNetV2	Swin-Tiny
SRe2L	15.41%	46.79%	36.59%	39.23%
G-VBSM	29.43%	51.82%	48.66%	57.40%

Table 2. ImageNet-1k Top-1 Acc. on cross-architecture generalization under IPC 50.

Cross-Architecture Generalization. The evaluation of cross-architecture generalization is crucial in assessing the quality of distilled datasets. Unlike traditional methods [1, 15, 20] which focus on CIFAR-100, our approach evaluates the effectiveness of the distilled dataset on ImageNet-1k, employing a suite of models with real-world applicability, including ResNet18 [5], MobileNetV2 [21], DeiT-Tiny [28], and Swin-Tiny [13]. The experimental results are reported in Table 2. From Tables 7 and 2, it is evident that the G-VBSM-synthetic dataset can effectively generalize across ResNet {18, 50, 101}, MobileNetV2, DeiT-Tiny, and Swin-Tiny architectures. Notably, DeiT-Tiny and Swin-Tiny, two architectures not encountered during the data synthesis phase, demonstrates significant proficiency with the accuracy 29.43% and 57.40%, outperforming the latest SOTA SRe2L by a margin of 14.01% and 18.17%, respectively.

Candidates (Backbone)				Evaluation Model	
ResNet18	MobileNetV2	EfficientNet-B0	ShuffleNetV2-0.5	ResNet18	ResNet50
✓				25.7%	30.2%
✓	✓			27.2%	31.2%
✓	✓	✓		27.9%	32.0%
✓	✓	✓	✓	31.4%	34.5%

Table 3. ImageNet-1k Top-1 Acc. about the number of candidate backbone in the soft label generation pipeline under IPC 10.

Ensemble in Soft Label Generation. In knowledge distillation, soft labels obtained through multiple teacher ensembles can effectively enhance the generalization of the student [35]. This observation is similarly corroborated in
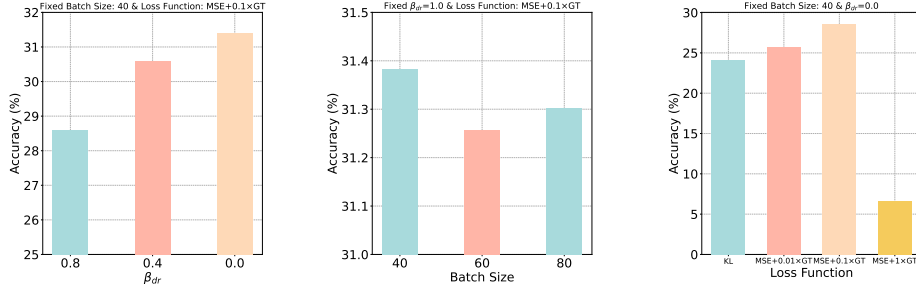
Figure 6. ImageNet-1k Top-1 Acc. of different loss function, β_{dr} and batch sizes under IPC 10.

Dataset	IPC	MTT [1] (CW128)	DataDAM [20] (CW128)	TESLA [2] (R18)	SRe2L (R18)	SRe2L (R50)	SRe2L (R101)	G-VBSM (R18)	G-VBSM (R50)	G-VBSM (R101)
Tiny-ImageNet	50	28.0±0.3	28.7±0.3	-	41.1±0.4	42.2±0.5	42.5±0.2	47.6±0.3	48.7±0.2	48.8±0.4
	100	-	-	-	49.7±0.3	51.2±0.4	51.5±0.3	51.0±0.4	52.1±0.3	52.3±0.1
ImageNet-1k	10	64.0±1.3†	6.3±0.0	7.7±0.1	21.3±0.6	28.4±0.1	30.9±0.1	31.4±0.5	35.4±0.8	38.2±0.4
	50	-	15.5±0.2	-	46.8±0.2	55.6±0.3	60.8±0.5	51.8±0.4	58.7±0.3	61.0±0.4
	100	-	-	-	52.8±0.3	61.0±0.4	62.8±0.2	55.7±0.4	62.2±0.3	63.7±0.2

Table 4. Comparison with baseline models in Tiny-ImageNet and ImageNet-1k. † indicates the ImageNette dataset, which contains only 10 classes. DataDAM [20] and TESLA [2] use the downsampled 64×64 ImageNet-1k. We cite the experimental results from SRe2L [34].

Dataset	IPC	Coreset Selection				Training Set Synthesis (CW128)							Training Set Synthesis (R18)		Whole Dataset
		Random	Herding	K-Center	Forgetting	DC [42]	DM [41]	CAFE [31]	KIP [15]	MTT [1]	DataDAM[20]	G-VBSM	SRe2L	G-VBSM	(CW128)
CIFAR-10	10	26.0±1.2	31.6±0.7	14.7±0.9	23.3±1.0	44.9±0.5	48.9±0.6	50.9±0.5	46.1±0.7	65.3±0.7	54.2±0.8	46.5±0.7	27.2±0.5	53.5±0.6	84.8±0.1
	50	43.4±1.0	40.4±0.6	27.0±1.4	23.3±1.1	53.9±0.5	63.0±0.4	62.3±0.4	53.2±0.7	71.6±0.2	67.0±0.4	54.3±0.3	47.5±0.6	59.2±0.4	
CIFAR-100	1	4.2±0.3	8.3±0.3	8.4±0.3	4.5±0.2	12.8±0.3	11.4±0.3	14.0±0.3	12.0±0.2	24.3±0.3	14.5±0.5	16.4±0.7	2.0±0.2	25.9±0.5	56.2±0.3
	10	14.6±0.5	17.3±0.3	17.3±0.3	15.1±0.3	25.2±0.3	29.7±0.3	31.5±0.2	40.1±0.4	33.1±0.4	34.8±0.5	38.7±0.2	31.6±0.5	59.5±0.4	
	50	30.0±0.4	33.7±0.5	30.5±0.3	-	30.6±0.6	43.6±0.4	47.7±0.2	-	42.9±0.3	49.4±0.3	45.7±0.4	49.5±0.3	65.0±0.5	

Table 5. Comparison with baseline models on CIFAR-10/100. All methods, except for SRe2L and G-VBSM, use a 128-width ConvNet (CW128) for data synthesis and evaluation. G-VBSM utilizes {CW128, WRN-16-2, ResNet18 (R18), ShuffleNetV2-0.5, MobileNetV2-0.5} for data synthesis and {CW128, R18} for evaluation. We cite the experimental results, except for SRe2L's, from DataDAM [20].

dataset distillation, as evidenced in Table 3. The greater the number of models, the stronger the generalization of the distilled dataset. It is particularly interesting to point out that ShuffleNetV2-0.5 improves further by more than 2% even though the other three models have already achieved modest ensemble gains. We attribute this enhancement to the channel shuffle mechanism within ShuffleNetV2-0.5, which imposes a beneficial regularization constraint for G-VBSM.

Logit Normalization	Evaluation Model			
	ResNet18 (MSE+0.1×GT)	ResNet18 (KL)	ResNet50 (MSE+0.1×GT)	ResNet50 (KL)
Yes	31.4%	25.4%	34.5%	28.6%
No	31.0%	25.1%	35.4%	31.9%

Table 6. ImageNet-1k Top-1 Acc. about the use of logit normalization under IPC 10.

Logit Normalization in Soft Label Generation. The aim of this strategy is to maintain the consistency of the logits (can be viewed as vectors) magnitude across all models within a high-dimensional space, thereby ensuring equal contribution of these models to the ultimate soft label for ensemble. This approach is not universally effective, as shown in Table 6. When the evaluation model parameter count is low (e.g., ResNet18), it can enhance G-VBSM performance. Conversely, with a model like ResNet50, it may hinder performance, rendering G-VBSM uncompetitive. As

a result, in this work, ResNet {50, 101} and ViT-based models do not employ logit normalization, while the remaining models do.

Loss Function in the Evaluation Phase. As illustrated in Fig. 6, replacing $D_{KL}(\cdot||\cdot)$ in SRe2L with MSE+γ×GT demonstrates to be extremely effective. For example, with β_{dr}=0.0 and batch size=40, adjusting γ to 0.01 and 0.1 enhances model performance during the evaluation phase by 1.4% and 6.0%, respectively. As a result, we employ a γ of 0.1 in all experiments conducted on ImageNet-1k. Furthermore, we observe that the distilled dataset's generalization is not significantly affected ($\leq 2\%$) by variations in either batch size or β_{dr}. This ensured that G-VBSM attain SOTA results with a minimal batch size 40 and β_{dr}=0.4 (i.e., the default settings in ImageNet-1k).
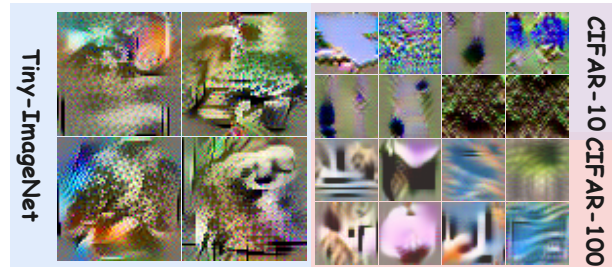

Figure 7. Synthetic data visualization on small-scale datasets.

Synthetic Data Visualization. The visualization results for the large-scale (the full 224×224 ImageNet-1k) and small-scale (CIFAR-10/100 and Tiny-ImageNet) datasets are displayed in Figs. 1 and 7, respectively. On ImageNet-1k, the images synthesized by G-VBSM are more informative and abstract compared with those from SRe2L. Furthermore, on small-scale datasets, the distilled images obtained by G-VBSM markedly differ from the images presented in MTT and DataDAM's papers [1, 20], illustrating that G-VBSM is an out-of-the ordinary algorithm. More synthetic data can be found in Appendix I.
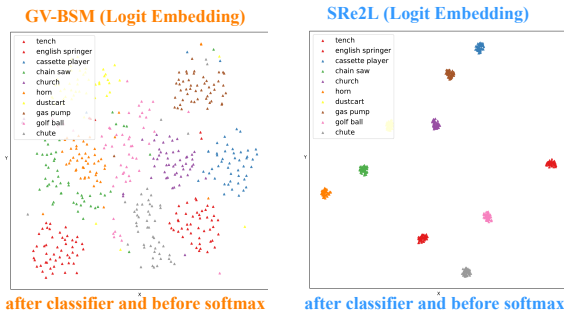

Figure 8. t-SNE visualization on ImageNet-1k.

Logit Embedding Distributions Visualization. We feed distilled datasets into a pre-trained model (*i.e.*, ResNet18) to obtain logit embeddings for t-SNE [29] visualization. Note that unlike previous methods that used feature embeddings before the classifier layer, we utilize logit embeddings after the classifier layer for presentation. Fig. 8 shows the distributions of G-VBSM and SRe2L logit embeddings. The logit embedding obtained by G-VBSM has less intra-class similarity than SRe2L, ensuring softer labels, preventing model overconfidence during the evaluation phase, and ultimately enhancing model generalization.

4.2. Large-Scale Dataset Comparison

Full 224×224 ImageNet-1k. As illustrated in the ImageNet-1k part of Table 4, G-VBSM consistently outperforms SRe2L across all IPC {10, 50, 100}, with the evaluation model being ResNet {18, 50, 101}, indicating that G-VBSM is highly competitive in the large-scale dataset ImageNet-1k. In particular, under extreme compression scenarios, such as IPC 10, G-VBSM achieves significant performance increases – 10.1%, 7.0%, and 7.3% for ResNet18, ResNet50, and ResNet101, respectively – compared with SRe2L. In comparison to the latest SOTA classical method, DataDAM, just experimenting on the full 64×64 ImageNet-1k, we exceed it by a margin of 23.7% under IPC 10. These results are particularly impressive when considered in the context of the large-scale dataset ImageNet-1k.

4.3. Small-Scale Dataset Comparison

Tiny-ImageNet. Similar to SRe2L, we evaluate the generalization of the distilled dataset on ResNet {18, 50, 101}. As presented in the Tiny-ImageNet part of Table 4, G-VBSM consistently surpasses the latest SOTA method SRe2L on different settings. Our method outperforms SRe2L, MTT and DataDAM by 6.5%, 19.6% and 18.9%, respectively, under IPC 50 applying 128-width ConvNet or ResNet18 as the evaluation model.

CIFAR-100. One of the disadvantages of SRe2L is its inability to compete with SOTA performance on this small-scale dataset benchmark (*i.e.*, evaluated with 128-width ConvNet and ResNet18), such as the performance on CIFAR-100 shown in Table 5, as a result, it fails to demonstrate that "local-match-global" (*e.g.*, statistical matching of BatchNorm) can rival or outperform traditional DC algorithms. In contrast, as shown in the CIFAR-100 part of Table 5, G-VBSM achieves accuracies 16.4% and 38.7%, the highest under IPC 1 and 10 on CIFAR-100 through "generalized matching", while maintaining the same number of epochs (*i.e.*, 1000) and the model (*i.e.*, 128-width ConvNet) for the evaluation phase as the traditional methods, outperforming the latest SOTA DataDAM by 1.9% and 3.9%, respectively. To the best of our knowledge, our proposed G-VBSM is the first algorithm that is strongly competitive on CIAR-100 as well as the full 224×224 ImageNet-1k.

CIFAR-10. As illustrated in the CIFAR-10 part of Table 5, G-VBSM outperforms vanilla DC [32], KIP [15], and all coreset selection algorithms on the benchmark with 128-width ConvNet as the evaluation model. Meanwhile, on the benchmark with ResNet 18 as the evaluation model, G-VBSM outperforms the *baseline* SRe2L by 26.3% and 11.7% under IPC 10 and 50, respectively.

5. Conclusion

In this paper, we introduce a novel perspective termed "generalized matching" for dataset condensation. This posits that an abundance of lightweight "local-match-global" matching are more effective than a single "local-match-global" matching, and even surpass the precise and costly matching used by traditional methods. Consequently, we present G-VBSM, designed to ensure data densification while performing matching based on sufficient and various backbones, layers, and statistics. Experiments conducted on CIFAR-10/100, Tiny-ImageNet, and the full 224×224 ImageNet-1k demonstrated that our method is the first algorithm to show strong performance across both small-scale and large-scale datasets. For future research, we plan to extend this approach's applicability to large-scale detection and segmentation datasets.

References

[1] George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A. Efros, and Jun-Yan Zhu. Dataset distillation by matching training trajectories. In *Computer Vision and Pattern Recognition*, New Orleans, LA, USA, 2022. IEEE. 2, 4, 6, 7, 8, 1, 3

[2] 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, Honolulu, Hawaii, USA, 2023. PMLR. 2, 7

[3] Jiawei Du, Yidi Jiang, Vincent Y. F. Tan, Joey Tianyi Zhou, and Haizhou Li. Minimizing the accumulated trajectory error to improve dataset distillation. In *Computer Vision and Pattern Recognition*, pages 3749–3758, Vancouver, BC, Canada, 2023. IEEE. 2, 4

[4] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021. 1

[5] Kaiming He, Xiangyu Zhang, and Shaoqing Ren. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition*, pages 770–778, Las Vegas, NV, USA, 2016. IEEE. 6

[6] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. 3, 1

[7] Shengyuan Hu, Jack Goetz, Kshitiz Malik, Hongyuan Zhan, Zhe Liu, and Yue Liu. Fedsynth: Gradient compression via synthetic data in federated learning. *arXiv preprint arXiv:2204.01273*, 2022. 2

[8] Tao Huang, Shan You, Fei Wang, Chen Qian, and Chang Xu. Knowledge distillation from a stronger teacher. In *Neural Information Processing Systems*, 2022. 3, 1

[9] 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, Baltimore, Maryland, USA, 2022. PMLR. 2

[10] Taehyeon Kim, Jaehoon Oh, NakYil Kim, Sangwook Cho, and Se-Young Yun. Comparing kullback-leibler divergence and mean squared error loss in knowledge distillation. In *International Joint Conference on Artificial Intelligence*, Virtual Event, 2021. Morgan Kaufmann. 5

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

[12] Yanqing Liu, Jianyang Gu, Kai Wang, Zheng Zhu, Wei Jiang, and Yang You. DREAM: efficient dataset distillation by representative matching. *arXiv preprint arXiv:2302.14416*, 2023. 2

[13] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *International Conference on Computer Vision*, pages 10012–10022, 2021. 6

[14] Wojciech Masarczyk and Ivona Tautkute. Reducing catastrophic forgetting with learning on synthetic data. In *Computer Vision and Pattern Recognition Workshops*, pages 252–253, Virtual Event, 2020. IEEE. 2

[15] Timothy Nguyen, Zhourong Chen, and Jaehoon Lee. Dataset meta-learning from kernel ridge-regression. *arXiv preprint arXiv:2011.00050*, 2020. 2, 6, 7, 8

[16] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Neural Information Processing Systems*, Vancouver, BC, Canada, 2019. 6, 2

[17] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *International Conference on Learning Representations*, 2023. 4, 2

[18] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 2, 5

[19] Noveen Sachdeva and Julian McAuley. Data distillation: A survey. *arXiv preprint arXiv:2301.04272*, 2023. 2

[20] Ahmad Sajedi, Samir Khaki, Ehsan Amjadian, Lucy Z Liu, Yuri A Lawryshyn, and Konstantinos N Plataniotis. Datadam: Efficient dataset distillation with attention matching. In *International Conference on Computer Vision*, pages 17097–17107, Paris, France, 2023. IEEE. 3, 6, 7, 8, 1

[21] Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Computer Vision and Pattern Recognition*, pages 4510–4520, Salt Lake City, UT, USA, 2018. IEEE. 6

[22] Mattia Sangermano, Antonio Carta, Andrea Cossu, and Davide Bacciu. Sample condensation in online continual learning. In *International Joint Conference on Neural Networks*, pages 1–8, Padua, Italy, 2022. IEEE. 2

[23] Zhiqiang Shen and Eric Xing. A fast knowledge distillation framework for visual recognition. In *European Conference on Computer Vision*, pages 673–690. Springer, 2022. 3

[24] Felipe Petroski Such, Aditya Rawal, Joel Lehman, Kenneth O. Stanley, and Jeffrey Clune. Generative teaching networks: Accelerating neural architecture search by learning to generate synthetic training data. In *International Conference on Machine Learning*, pages 9206–9216, Virtual Event, 2020. PMLR. 2

[25] Mingxing Tan and Quoc Le. Efficientnetv2: Smaller models and faster training. In *International Conference on Machine Learning*, pages 10096–10106, Virtual Event, 2021. PMLR. 6

[26] Amirhossein Tavanaei. Embedded encoder-decoder in convolutional networks towards explainable AI. 2020. 5

[27] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. In *International Conference on Learning Representations*, 2019. 1

[28] 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, Virtual Event, 2021. PMLR. 6

[29] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9 (11), 2008. 8

[30] Chunwei Wang, Chao Ma, Ming Zhu, and Xiaokang Yang. Pointaugmenting: Cross-modal augmentation for 3d object detection. In *Computer Vision and Pattern Recognition*, pages 11794–11803, Virtual Event, 2021. 2

[31] 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 *Computer Vision and Pattern Recognition*, pages 12196–12205, New Orleans, LA, USA, 2022. IEEE. 2, 3, 7

[32] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A Efros. Dataset distillation. *arXiv preprint arXiv:1811.10959*, 2018. 1, 2, 8, 3

[33] Hongxu Yin, Pavlo Molchanov, Jose M Alvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *Computer Vision and Pattern Recognition*, pages 8715–8724, Virtual Event, 2020. IEEE. 2, 5

[34] Zeyuan Yin, Eric P. Xing, and Zhiqiang Shen. Squeeze, recover and relabel: Dataset condensation at imagenet scale from A new perspective. In *Neural Information Processing Systems*. NIPS, 2023. 2, 3, 4, 5, 6, 7, 1

[35] Shan You, Chang Xu, Chao Xu, and Dacheng Tao. Learning from multiple teacher networks. In *International Conference on Knowledge Discovery and Data Mining*, pages 1285–1294, Halifax, NS, Canada, 2017. ACM. 6

[36] Ruonan Yu, Songhua Liu, and Xinchao Wang. Dataset distillation: A comprehensive review. *arXiv preprint arXiv:2301.07014*, 2023. 2

[37] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *British Machine Vision Conference*, York, UK, 2016. BMVA Press. 1, 6

[38] 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 *Computer Vision and Pattern Recognition*, Vancouver, BC, Canada, 2023. IEEE. 6, 5

[39] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Computer Vision and Pattern Recognition*, pages 6848–6856, 2018. 6

[40] Bo Zhao and Hakan Bilen. Dataset condensation with differentiable siamese augmentation. In *International Conference on Machine Learning*, pages 12674–12685, Virtual Event, 2021. PMLR. 2

[41] Bo Zhao and Hakan Bilen. Dataset condensation with distribution matching. In *Winter Conference on Applications of Computer Vision*, pages 6514–6523, Waikoloa, Hawaii, 2023. IEEE. 2, 7, 5

[42] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. Dataset condensation with gradient matching. In *International Conference on Learning Representations*, Virtual Event, 2021. OpenReview.net. 2, 7

[43] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *Computer Vision and Pattern Recognition*, pages 11953–11962, 2022. 1

[44] Yongchao Zhou, Ehsan Nezhadarya, and Jimmy Ba. Dataset distillation using neural feature regression. In *Neural Information Processing Systems*, New Orleans, LA, USA, 2022. NIPS. 2