PSM: DATASET DISTILLATION BASED ON PRECISE STATISTICAL MATCHING BY DIFFICULTY

Anonymous authors
Paper under double-blind review

ABSTRACT

Dataset distillation (DD) condenses a large original set into a small distilled set with high training utility. Decoupled statistical matching methods substantially reduce distillation time and memory overhead while achieving strong performance. However, they typically supervise all distilled data with running statistics estimated from the entire original set. These statistics mainly capture the average feature distribution while ignoring the difficulty differences among original data, making them insufficient for characterizing the difficulty structure of the original data. To address this issue, we propose **P**recise **S**tatistical **M**atching (**PSM**) by difficulty. After pretraining, PSM uses the Global Precision Score (GPS) to evaluate image difficulty, sorts the original data according to the scores, and then partitions it into IPC (Images Per Class of distilled data) difficulty groups. During distillation, Statistics Updated Again (SUA) updates the teacher's BN running statistics using original data from each group via forward passes, providing precise supervision with corresponding difficulty for distilled data at each id_{IPC}. Meanwhile, Initial Sample Screening (ISS) initializes distilled data with original images from the corresponding group, providing an effective starting point for precise matching. Experiments across multiple datasets and model architectures demonstrate the effectiveness of PSM, showing that it broadens the difficulty range of distilled samples and further improves the final performance. Code will be released.

1 INTRODUCTION

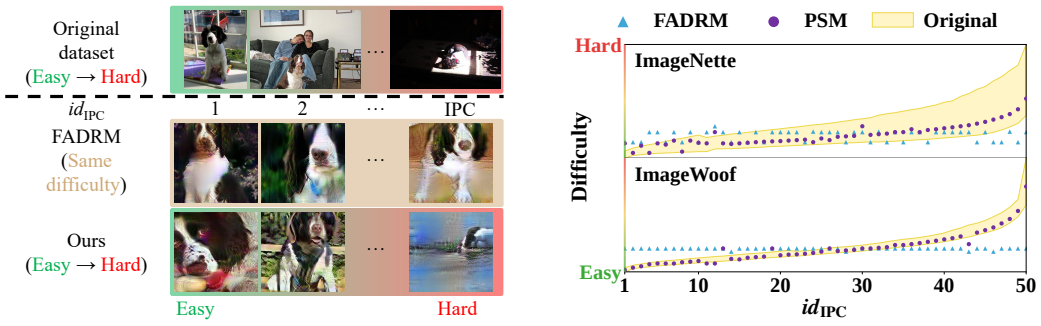
As datasets grow and neural networks advance, the computational and storage resources required for models have increased substantially (Hoffmann et al., 2022; Sangermano et al., 2022), becoming a major limitation on model development. Dataset distillation (DD) aims to distill a large original set into a small set with high training utility (Lei & Tao, 2024; Liu & Du, 2025), such that models trained on the distilled set can achieve performance comparable to those trained on the original set.

Representative DD methods include gradient matching (Zhao & Bilen, 2021), distribution matching (Li et al., 2026), trajectory matching (Zhong et al., 2025a), generative methods (Gan et al., 2026), and decoupled methods (Shang et al., 2026). Matching methods often rely on bi-level optimization, while generative methods require large generative models, incurring substantial computational and time costs. Recently, decoupled methods (Yin et al., 2023) based on BN (Batch Normalization) statistics matching (Ioffe & Szegedy, 2015) have received increasing attention. By separating teacher pretraining from data distillation, and using the teacher's BN running statistics to supervise distilled data, they markedly reduce these costs while retaining strong performance.

However, existing decoupled methods typically apply BN running statistics estimated from the entire original set to all distilled data. These global statistics mainly characterize the average feature distribution; uniformly matching them therefore smooths out the representation differences between easy and hard data. As shown in Figure 1a, the data distilled by FADRM exhibit visually similar difficulty, and Figure 1b further reveals that their difficulty remains nearly constant. Consequently, the distilled data cannot adequately cover the range of feature distributions spanned by original data of varying difficulty, limiting the student's ability to learn the difficulty structure of the original data.

To address this limitation, we propose **P**recise **S**tatistical **M**atching (**PSM**) by difficulty. To align difficulty estimation with statistical matching in the same feature space, we carefully design the Global Precision Score (GPS): after pretraining, for each original image, GPS computes statistics at the inputs to the teacher's BN layers, and then measures image difficulty as their distance from the corresponding BN running statistics (under the ensemble-based paradigm, PSM averages the GPS

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) (b)

Figure 1: (a) Visualizations for the English Springer class in the original ImageNette dataset and the distilled datasets. The original data exhibit a progression in difficulty from easy to hard. Ours effectively preserves this difficulty structure, whereas the data distilled by FADRM show limited variation in difficulty. (b) Difficulty distributions of original and distilled data on Imagenette/Woof. FADRM exhibits nearly constant difficulty across id_{IPC}, indicating limited variation. In contrast, ours increases with id_{IPC} in line with the original data and covers a broader difficulty range.

rankings of all teachers to integrate their judgments); a higher GPS indicates greater deviation from the average feature distribution, and thus greater difficulty; based on the GPS ranking, PSM then partitions the original data within each class into IPC (Images per Class) difficulty groups.

During distillation, when optimizing the g-th batch of distilled data, Statistics Updated Again (SUA) feeds original data from the g-th group through the teacher to update its BN running statistics via forward propagation, producing supervision that more accurately characterizes the current difficulty group. Initial Sample Screening (ISS) selects real images from the same group to initialize the distilled data, aligning their initial difficulty with the target group. Together, SUA and ISS construct statistical supervision and initialization for different difficulty groups, enabling the distilled data to cover a broader difficulty range, and helping the student to learn a more complete difficulty structure.

Extensive experiments on multiple datasets demonstrate that PSM outperforms existing state-of-the-art (SOTA) methods, validating its effectiveness. Our contributions are summarized as follows:

- We revisit the existing decoupled DD paradigm, in which BN running statistics estimated from the original set serve as supervision signals, and show that this paradigm limits variation in difficulty within the distilled data. To address this issue, we propose GPS, a difficulty metric closely aligned with statistical supervision, to rank the original data by difficulty.
- We propose PSM, a method that broadens the range of difficulty covered by the distilled data by adjusting statistical supervision and initial data: SUA updates the teacher model's BN running statistics through forward propagation; ISS selects original images from the corresponding difficulty group to initialize the distilled data.
- We conducted experiments on multiple datasets and model architectures, and the results show that PSM broadens the difficulty range of distilled samples, further improving the final performance and demonstrating its effectiveness.

2 Related Work

Dataset distillation (DD) compresses a large dataset into a small set of samples with high training utility, reducing storage and training costs while retaining performance comparable to training on the full dataset. Existing methods mainly include gradient matching (Lee et al., 2022; Liu et al., 2023), distribution matching (Xue et al., 2025; Li et al., 2025), trajectory matching (Chen et al., 2023; Guo et al., 2024), decoupled distillation (Yin & Shen, 2024; Xiao & He, 2024), and generative distillation (Zhong et al., 2025b; Cai et al., 2026). SRe^{2}L (Yin et al., 2023) pioneered decoupled distillation by separating teacher pretraining from distillation. FADRM (Cui et al., 2025a) employs multiscale data residual connections to preserve original image information and enrich distilled samples, while FADRM+ extends it to multiple teachers. Further related work is discussed in the appendix A.

3 PRELIMINARIES

3.1 DATASET DISTILLATION

Let $\mathcal{T} = \{(x_i, y_i)\}_{i=1}^{N}$ denote the original set, and let $\mathcal{D} = \{(\tilde{x}_j, \tilde{y}_j)\}_{j=1}^{M}$ denote the distilled set, where $M \ll N$. Let $\theta_{\mathcal{T}}$ and $\theta_{\mathcal{D}}$ represent the parameters obtained by training on $\mathcal{T}$ and $\mathcal{D}$, respectively. DD aims to construct $\mathcal{D}$ such that models trained on $\mathcal{D}$ achieve performance comparable to those trained on $\mathcal{T}$. Given a distillation budget M, this objective can be formulated as

$$\mathcal{D}^* = \underset{\mathcal{D}:\,|\mathcal{D}|=M}{\arg\min} \sup_{(x,y)\in\mathcal{T}} \left|\mathcal{L}(f_{\theta_{\mathcal{T}}}(x), y) - \mathcal{L}(f_{\theta_{\mathcal{D}}}(x), y)\right|, \tag{1}$$

where $\mathcal{L}(\cdot,\cdot)$ denotes the task loss. Under a given distillation budget, the number of images per class (IPC) is the same across all classes. Typically, a student trained on the distilled set is evaluated on the original test set, and its test performance is used to measure the quality of the distilled set.

3.2 DECOUPLED DATASET DISTILLATION

SRe^{2}L (Yin et al., 2023) introduced a three-stage framework that separates model training from data distillation. During distillation, the pretrained teacher is fixed, and only the distilled data are optimized, avoiding the inner optimization of model parameters and reducing computational overhead.

I. Teacher model pretraining. The teacher is trained on the original set $\mathcal{T}$ to learn feature representations, while its BN layers accumulate running means and variances from the training data. The resulting parameters $\theta_{\mathcal{T}}$ and BN statistics are then fixed to guide distillation.

II. Distillation. A distilled batch $\widetilde{\mathcal{B}} = (\tilde{x}_j, \tilde{y}_j)_{j=1}^{B}$ is fed into the frozen teacher, where B is typically equal to the number of classes C. The classification loss $\mathcal{L}_{\text{cls}}$ enforces semantic consistency, while the statistics matching loss $\mathcal{L}_{\text{BN}}$ aligns the batch statistics with the teacher's running statistics:

$$\begin{aligned} \mathcal{L}_{\text{dist}}(\widetilde{\mathcal{B}}) &= \mathcal{L}_{\text{cls}}\left(\tilde{x}, \tilde{y}\right) + \lambda_{\text{BN}}\mathcal{L}_{\text{BN}}\left(\{\tilde{x}_j\}_{j=1}^{B}\right), \\ \text{where} \quad \mathcal{L}_{\text{cls}}\left(\tilde{x}, \tilde{y}\right) &= \mathbb{E}_{(\tilde{x},\tilde{y})\sim\widetilde{\mathcal{B}}}\left[\mathcal{L}_{\text{CE}}\left(f_{\theta_{\mathcal{T}}}(\tilde{x}), \tilde{y}\right)\right], \\ \mathcal{L}_{\text{BN}}\left(\{\tilde{x}_j\}_{j=1}^{B}\right) &= \textstyle\sum_{\ell=1}^{L}\left(\|\mu_\ell - \bar{\mu}_\ell\|_2 + \left\|\sigma_\ell^2 - \bar{\sigma}_\ell^2\right\|_2\right), \end{aligned} \tag{2}$$

where $\mathcal{L}_{\text{CE}}$ is the cross entropy loss; λ_{BN} weights statistics matching; and L is the number of BN layers, indexed by ℓ. For a BN layer with C_ℓ channels, $\mu_\ell, \sigma_\ell^2 \in \mathbb{R}^{C_\ell}$ are the batch mean and variance at its input, while $\bar{\mu}_\ell, \bar{\sigma}_\ell^2 \in \mathbb{R}^{C_\ell}$ are the corresponding teacher running statistics.

III. Soft label generation. The distilled data are fed into the pretrained teacher, whose class probability distributions are used as soft labels. These labels also preserve semantic relationships for all classes. The student is then trained on the distilled data and soft labels, receiving richer supervision.

3.3 BATCH NORMALIZATION

Batch Normalization (BN) (Ioffe & Szegedy, 2015) was originally introduced to mitigate changes in intermediate feature distributions. For features entering a BN layer, BN first computes the mean and variance of the current batch for each channel, and uses them to normalize the features. This process stabilizes feature scales across layers and improves optimization stability.

During training, each BN layer uses the current batch statistics for feature normalization, and its running mean and variance are continuously updated through an exponential moving average. Given the training batch $\mathcal{B}_t$ at step t, the running statistics of BN layer ℓ are updated as follows:

$$\bar{\mu}_{\ell,t} = (1-\rho)\bar{\mu}_{\ell,t-1} + \rho\mu_\ell(\mathcal{B}_t), \qquad \bar{\sigma}^2_{\ell,t} = (1-\rho)\bar{\sigma}^2_{\ell,t-1} + \rho\sigma^2_\ell(\mathcal{B}_t), \tag{3}$$

where $\rho \in (0, 1]$ controls the statistics update, a larger ρ assigns more weight to the current batch.

4 METHOD

4.1 LIMITATIONS OF DECOUPLED DATASET DISTILLATION

SRe^{2}L (Yin et al., 2023) pioneered the decoupled DD paradigm, which uses the BN running statistics stored in a pretrained teacher as supervisory signals to guide the optimization, as shown in equation 2. Because of its efficiency, this paradigm has attracted considerable attention. Recent studies have improved this paradigm by reducing storage overhead (Cui et al., 2026), lowering computational complexity (Shang et al., 2026), and enhancing performance (Shi et al., 2026). However, most

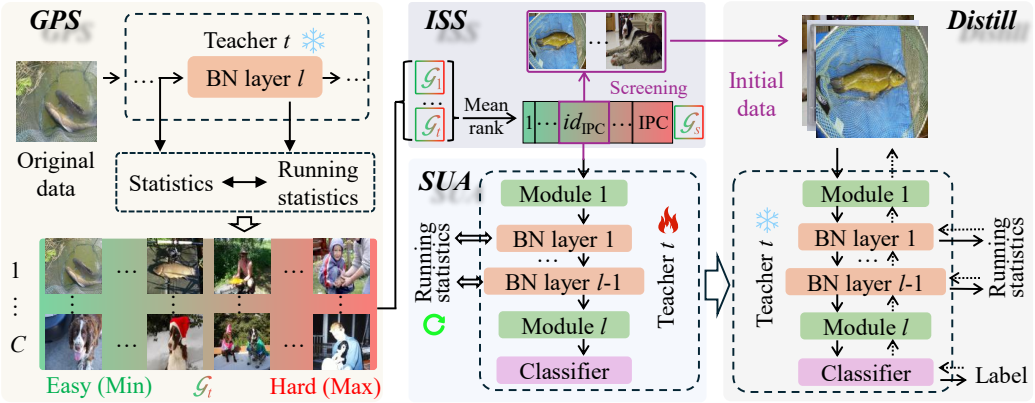


Figure 2: Overview of the PSM. First, GPS evaluates the difficulty of the original data and ranks them from easy to hard within each class. The ordered data are then divided into IPC difficulty groups. When distilling the g-th batch, SUA feeds the original data in difficulty group g into the teacher for forward propagation, allowing the statistics of each BN layer to characterize the data features. Meanwhile, ISS selects original data from the same difficulty group to initialize the corresponding distilled data, thereby providing a favorable starting point for subsequent optimization.

of these studies rely on additional components or optimization mechanisms, with limited attention paid to the fundamental supervisory signals and their influence.

Homogeneous Supervision. Existing decoupled methods use the global BN running statistics stored in the pretrained teacher as the target for every distilled batch. Let $\boldsymbol{\tau}_g$ denote the statistical supervision assigned to batch g, and let $\boldsymbol{\tau}^{\mathrm{G}}$ denote the global BN statistics. Then

$$\tau_g = \tau^{\mathrm{G}},\ \forall g \quad \Longrightarrow \quad \mathcal{D}_{\mathrm{sup}}(\tau_g, \tau_h) = 0,\ \forall g, h.$$

Consequently, the statistical target is independent of the batch index, and cannot explicitly provide batch-specific difficulty supervision. Although initialization and optimization may introduce difficulty variation, this variation is not controlled by the supervision signal, making it difficult to preserve the difficulty structure of the original data, as illustrated in Figure 1b.

4.2 Dataset Distillation Based on Precise Statistical Matching by Difficulty

Figure 2 illustrates the overall pipeline of the proposed method. Given the original set and a pretrained teacher, we first employ GPS to evaluate the difficulty of each original data point and rank the data within each class from easy to hard. We then divide the ordered data of each class into IPC difficulty groups, with different groups representing data of varying difficulty in the original set. When distilling the g-th batch, SUA feeds the original data in difficulty group g into the teacher to further adapt its BN running statistics to the characteristics of the current group, thereby constructing distinct statistical supervision signals for the corresponding distilled batch. By matching the respective statistical supervision signals, the distilled data in different batches progressively learn original data patterns from easy to hard. Meanwhile, ISS selects original data from the corresponding difficulty group to initialize the distilled data, aligning the difficulty of the initial data with that of the target distilled batch, and providing a favorable starting point for subsequent optimization. GPS, SUA, and ISS are introduced in the following sections. See the appendix D for pseudocode.

4.2.1 GPS: Global Precision Score

To accurately partition the original set by difficulty and ensure that difficulty estimation and BN statistics operate in the same feature space, we propose a new difficulty partitioning method, GPS.

Operation and Purpose. We compute the per-channel mean and variance of x_i at the input to each BN layer, and compare them with the corresponding running mean and variance. GPS is defined as

$$\mathrm{GPS}(x_i) = \mathbb{E}_{k\in\{1,\dots,K\}}\left[\mathrm{rank}_c\left(\mathbb{E}_{\ell\in\{1,\dots,L_k\}}\left[\frac{\|\boldsymbol{\mu}_{\ell,i,k} - \bar{\boldsymbol{\mu}}_{\ell,k}\|_2 + \beta\left\|\log\frac{\boldsymbol{\sigma}^2_{\ell,i,k}+\epsilon}{\bar{\boldsymbol{\sigma}}^2_{\ell,k}+\epsilon}\right\|_2}{\sqrt{C_{\ell,k}}}\right]\right)\right]. \tag{4}$$

Here K denotes the number of teachers, $\mathrm{rank}\, c(\cdot)$ is the rank of a sample within class c, β balances the mean and variance distances, and ϵ ensures numerical stability. For the variance, we first compute the ratio between the sample variance and the BN running variance to reduce the effect of variance scale differences across channels. We then take the logarithm of this ratio to stabilize its numerical range. Finally, we normalize the statistical distance at each layer by $\sqrt{C_{\ell,k}}$ to reduce distance scale discrepancies caused by the number of channels across BN layers.

A larger GPS indicates that the data exhibits more pronounced statistical deviations across multiple feature layers, and thus corresponds to greater difficulty. Based on the GPS ranking, PSM divides the ordered data of each class into IPC difficulty groups of approximately equal size, which sequentially represent the original data from easy to hard. The resulting difficulty groups provide the basis for SUA to construct distinct supervision, and for ISS to select the corresponding initialization data.

Empirical Observation. A larger GPS indicates more pronounced deviations between the sample statistics and the teacher's BN running statistics. To verify the correlation between statistics deviations and classification difficulty, we sort the original samples in ascending order of GPS, and compute the teacher's cross-entropy loss on each sample's ground truth label. The empirical relationship between GPS and classification difficulty is expressed as

$$\mathrm{GPS}(x_i)\uparrow \quad \Longrightarrow \quad \mathcal{L}_{\mathrm{CE}}(f_\theta(x_i), y_i) = -\log p_\theta(y_i \mid x_i)\uparrow . \tag{5}$$

Experimental results show that the cross-entropy loss generally increases along the GPS ranking. Samples with larger GPS values tend to have higher classification losses. These results indicate that GPS effectively reflects the sample difficulty perceived by the teacher, providing a basis for grouping samples from easy to hard. A detailed analysis is provided in the appendix B.1.

4.2.2 SUA: Statistics Updated Again

To convert the difficulty groups obtained by GPS into statistical supervision corresponding to individual distillation batches, we propose SUA.

Operation and Purpose. Let $\mathcal{T}_g$ denote the g-th difficulty group. We set IPC difficulty groups, with the g-th group corresponding to the id_{IPC}-th distillation batch. For teacher k, SUA starts from the global BN running statistics $\boldsymbol{\tau}_k^{\mathrm{G}}$ stored in the pretrained teacher, freezes the teacher parameters θ_k, and performs $\lfloor r_{\mathrm{fwd}} E_k^{\mathrm{pre}} \rfloor$ epochs of forward propagation over $\mathcal{T}_g$. During this process, only the BN running statistics are updated according to equation 3. This process is expressed as

$$\boldsymbol{\tau}_k^{\mathrm{G}} \xrightarrow{f_{\theta_k=\mathrm{constant}}^{\lfloor r_{\mathrm{fwd}} E_k^{\mathrm{pre}} \rfloor}(\mathcal{T}_g)} \boldsymbol{\tau}_{k,g}^{\mathrm{SUA}}. \tag{6}$$

where r_{fwd} denotes the ratio of forward propagation epochs, E_k^{pre} denotes the number of pretraining epochs for teacher k, $\lfloor r_{\mathrm{fwd}} E_k^{\mathrm{pre}} \rfloor$ denotes the actual number of forward propagation epochs, and $\boldsymbol{\tau}_{k,g}^{\mathrm{SUA}}$ denotes the BN running statistics obtained by SUA.

Through this update, the BN running statistics of the teacher gradually approach those of the current difficulty group, and serve as the statistical matching signal for distillation batch g, thereby providing each distillation batch with supervision signal corresponding to its difficulty.

Proposition 1 (Expected Convergence; Proof in Appendix B.2)**.** *For any fixed teacher k, let $\boldsymbol{\tau}_{k,g}^{\star}$ and $\boldsymbol{\tau}_{k,g}^{(t)}$ denote the expected BN batch statistics of $\mathcal{T}_g$ and the running statistics after t forward batch updates, respectively. If the forward batches are independently drawn from $\mathcal{T}_g$, then*

$$\mathbb{E}\left[\boldsymbol{\tau}_{k,g}^{(t)}\right] - \boldsymbol{\tau}_{k,g}^{\star} = (1-\rho)^t \left(\boldsymbol{\tau}_k^{\mathrm{G}} - \boldsymbol{\tau}_{k,g}^{\star}\right) \xrightarrow{t\to\infty} \mathbf{0}. \tag{7}$$

Thus, SUA statistics converge in expectation to those of the current difficulty group. The updated statistics then serve as the matching signal for batch g, guiding its distilled data to match the corresponding feature statistics.

4.2.3 ISS: Initial Sample Screening

The initialization of distilled data significantly affects final performance. EDC (Shao et al., 2024b) shows that real image initialization reduces the distribution gap between the initialized and original data. SelMatch (Lee & Chung, 2024) further shows that real samples of appropriate difficulty help introduce corresponding complex features. Motivated by these findings, we propose ISS to initialize each distillation batch with real data matching its target difficulty.

Specifically, let $\mathcal{T}_{g,c}$ denote the original data of class c in difficulty group g. ISS further selects real images from $\mathcal{T}_{g,c}$ based on their difficulty, and uses them to initialize the distilled data $\widetilde{x}_{g,c}^{(0)}$. Since the initialization images and the statistical supervision come from the same difficulty group, they are aligned in difficulty, reducing the statistical bias at the beginning of optimization. Experiments show that ISS significantly improves final distillation performance.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Baselines. FADRM (Cui et al., 2025a) is a recent SOTA method for decoupled DD, achieving improved performance while substantially reducing memory and time costs, we therefore adopt it as our primary baseline and further compare with FADRM+, its extension using multiple models. SRe^{2}L (Yin et al., 2023), a representative foundational method for decoupled DD, is also included for comparison. In addition, we include RDED (Sun et al., 2024) as a representative method from other DD paradigms, it constructs distilled data by assembling real images, which has been adopted by other DD methods for distilled data initialization.

Datasets. We evaluate PSM on benchmark datasets widely used in dataset distillation. For low resolution datasets, we use CIFAR-10/100 (Krizhevsky, 2009) at 32×32, and Tiny-ImageNet (Le & Yang, 2015) at 64×64. For high-resolution datasets, we mainly use ImageNet-1K (Deng et al., 2009) at 224×224 and its subsets, ImageWoof (fine-grained dataset) and ImageNette.

Implementation Details. For PSM, we follow the post-evaluation protocol of FADRM. Students are trained for 1000 epochs on Tiny-ImageNet with $\mathrm{IPC} = 1$ and CIFAR-10/100, and for 300 epochs in all other settings. For a fair comparison with FADRM+, we further adopt the multi-teacher variant PSM+, which consistently employs ShuffleNetV2 (Ma et al., 2018), ResNet18 (He et al., 2016), MobileNetV2 (Sandler et al., 2018), and DenseNet121 (Huang et al., 2017) as teachers for distillation across all datasets. The remaining parameters are kept consistent with FADRM, while other baselines follow their original settings. See the appendix D for more details.

5.2 COMPARISON WITH STATE-OF-THE-ART METHODS

5.2.1 MAIN RESULTS

In Table 1, methods marked with "+" use multiple teachers for distillation and a single student for evaluation, while the remaining methods follow the standard same-architecture evaluation protocol.

Low-resolution datasets. As shown in Table 1, we evaluate PSM on CIFAR-10/100 and Tiny-ImageNet. The results show that PSM outperforms its corresponding baselines in most settings when evaluated with ResNet18/50. Notably, on CIFAR-10 with $\mathrm{IPC} = 10$, PSM achieves a Top-1 accuracy of 52.7% with ResNet50, outperforming FADRM by 8.9%, and demonstrating the effectiveness of PSM. On CIFAR-10 with $\mathrm{IPC} = 1$, PSM performs comparably to its corresponding baselines. This may be mainly attributed to CIFAR-10 containing only 10 coarse categories, for which the global statistics may already adequately characterize the overall difficulty distribution. Consequently, SUA induces only limited changes in the statistics, resulting in small performance differences.

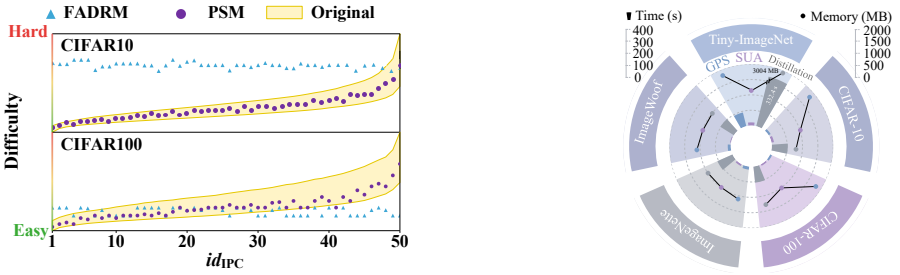
High-resolution datasets. We evaluate PSM on ImageNette, ImageWoof, and ImageNet-1K. As shown in Table 1, PSM achieves consistent performance gains in most settings, demonstrating its strong robustness. Performance degradation occurs mainly on ImageWoof with ResNet50 as the student under the $\mathrm{IPC} = 1/10$. One possible explanation is the limited compatibility of ResNet50 with ImageWoof: when trained on the original set, ResNet18 achieves an accuracy of 85.0%, whereas ResNet50 achieves only 80.3%. Under low IPC, the limited distilled data may further amplify this mismatch, making it difficult for ResNet50 to learn the difficulty information preserved by PSM.

5.2.2 DIFFICULTY AND EFFICIENCY ANALYSIS.

Difficulty. Figure 3a shows the difficulty distributions on CIFAR-10/100. Together with Figure 1b, these results show that the difficulty of the data distilled by PSM follows a trend consistent with that of the original data and spans a broad range, whereas the difficulty of the data distilled by FADRM remains nearly constant, exhibiting limited variation. These results demonstrate that PSM effectively preserves the difficulty structure of the original data across datasets of different scales and resolu-

Table 1: Top-1 accuracy comparison of PSM, extended variant PSM+, and other SOTA methods. For each dataset, IPC, and student setting, the mean values of the **best** and second best results are marked in **bold** and underlined, respectively. Values in parentheses after the PSM and PSM+ results indicate their performance changes relative to FADRM and FADRM+, respectively.

Student		ResNet18						ResNet50 (He et al., 2016)					
Dataset	IPC	SRe^{2}L	RDED	FADRM	PSM	FADRM+	PSM+	SRe^{2}L	RDED	FADRM	PSM	FADRM+	PSM+
CIFAR-10	1	16.6$_{\pm 0.9}$	22.9$_{\pm 0.4}$	19.3$_{\pm 0.6}$	20.9$_{\pm 0.1}$ (↑ 1.6)	**23.7**$_{\pm 0.8}$	22.3$_{\pm 0.6}$ (↓ 1.4)	15.2$_{\pm 1.3}$	19.7$_{\pm 1.7}$	23.2$_{\pm 0.7}$	23.2$_{\pm 0.2}$ (= 0.0)	**23.5**$_{\pm 1.3}$	23.3$_{\pm 0.4}$ (↓ 0.2)
	10	29.3$_{\pm 0.5}$	37.1$_{\pm 0.3}$	48.2$_{\pm 0.4}$	54.5$_{\pm 1.3}$ (↑ 6.3)	55.9$_{\pm 1.0}$	**61.1**$_{\pm 0.8}$ (↑ 5.2)	30.3$_{\pm 1.7}$	32.5$_{\pm 0.9}$	43.8$_{\pm 1.1}$	52.7$_{\pm 0.1}$ (↑ 8.9)	55.1$_{\pm 0.8}$	**56.8**$_{\pm 1.1}$ (↑ 1.7)
	50	45.0$_{\pm 0.7}$	62.1$_{\pm 0.1}$	80.6$_{\pm 0.7}$	81.1$_{\pm 0.1}$ (↑ 0.5)	85.8$_{\pm 0.1}$	**86.7**$_{\pm 0.2}$ (↑ 0.9)	52.9$_{\pm 1.3}$	52.5$_{\pm 2.0}$	79.7$_{\pm 1.2}$	82.7$_{\pm 0.7}$ (↑ 3.0)	83.7$_{\pm 0.6}$	**86.0**$_{\pm 0.9}$ (↑ 2.3)
	Whole dataset				92.9						93.1		
CIFAR-100	1	6.6$_{\pm 0.2}$	11.1$_{\pm 0.3}$	31.3$_{\pm 0.2}$	32.9$_{\pm 0.9}$ (↑ 1.6)	37.9$_{\pm 0.8}$	**39.1**$_{\pm 0.1}$ (↑ 1.2)	6.0$_{\pm 0.1}$	11.6$_{\pm 0.4}$	24.4$_{\pm 1.6}$	26.2$_{\pm 1.4}$ (↑ 1.8)	33.6$_{\pm 0.8}$	**34.9**$_{\pm 1.3}$ (↑ 1.3)
	10	27.0$_{\pm 0.4}$	42.6$_{\pm 0.2}$	64.7$_{\pm 0.4}$	**66.7**$_{\pm 0.4}$ (↑ 2.0)	62.5$_{\pm 0.2}$	66.1$_{\pm 0.2}$ (↑ 3.6)	35.4$_{\pm 1.9}$	50.3$_{\pm 0.4}$	61.8$_{\pm 0.5}$	64.6$_{\pm 0.1}$ (↑ 2.8)	62.3$_{\pm 0.2}$	**65.7**$_{\pm 0.4}$ (↑ 3.4)
	50	50.2$_{\pm 0.4}$	62.6$_{\pm 0.1}$	69.3$_{\pm 0.3}$	69.7$_{\pm 0.3}$ (↑ 0.4)	67.4$_{\pm 0.4}$	**70.1**$_{\pm 0.1}$ (↑ 2.7)	52.9$_{\pm 0.1}$	66.8$_{\pm 0.3}$	67.6$_{\pm 0.5}$	69.8$_{\pm 0.1}$ (↑ 2.2)	68.2$_{\pm 0.1}$	**70.7**$_{\pm 0.2}$ (↑ 2.5)
	Whole dataset				73.6						74.4		
Tiny-ImageNet	1	2.6$_{\pm 0.1}$	9.7$_{\pm 0.4}$	28.6$_{\pm 0.1}$	30.6$_{\pm 1.3}$ (↑ 2.0)	32.4$_{\pm 0.2}$	**35.4**$_{\pm 0.7}$ (↑ 3.0)	5.1$_{\pm 0.1}$	6.5$_{\pm 0.5}$	31.1$_{\pm 0.4}$	31.3$_{\pm 0.7}$ (↑ 0.2)	31.0$_{\pm 1.3}$	**32.9**$_{\pm 0.9}$ (↑ 1.9)
	10	16.1$_{\pm 0.2}$	41.9$_{\pm 0.2}$	46.5$_{\pm 0.4}$	48.1$_{\pm 0.3}$ (↑ 1.6)	47.3$_{\pm 1.1}$	**48.4**$_{\pm 0.3}$ (↑ 1.1)	43.0$_{\pm 0.5}$	36.9$_{\pm 0.4}$	47.5$_{\pm 0.3}$	48.0$_{\pm 0.3}$ (↑ 0.5)	47.1$_{\pm 0.4}$	**48.2**$_{\pm 0.9}$ (↑ 1.1)
	50	41.1$_{\pm 0.4}$	**58.2**$_{\pm 0.1}$	51.4$_{\pm 0.1}$	51.7$_{\pm 0.1}$ (↑ 0.3)	52.0$_{\pm 1.0}$	52.1$_{\pm 0.1}$ (↑ 0.1)	58.3$_{\pm 0.1}$	48.0$_{\pm 0.8}$	**57.8**$_{\pm 0.1}$	56.6$_{\pm 0.7}$ (↓ 1.2)	52.2$_{\pm 1.8}$	53.0$_{\pm 0.2}$ (↑ 0.8)
	Whole dataset				60.3						63.4		
ImageNette	1	19.1$_{\pm 1.1}$	**35.8**$_{\pm 1.0}$	28.2$_{\pm 0.6}$	25.0$_{\pm 0.5}$ (↓ 3.2)	30.5$_{\pm 1.3}$	32.5$_{\pm 0.5}$ (↑ 2.0)	13.6$_{\pm 0.5}$	23.6$_{\pm 1.4}$	23.7$_{\pm 0.4}$	22.9$_{\pm 0.8}$ (↓ 0.8)	25.5$_{\pm 0.7}$	**29.2**$_{\pm 0.6}$ (↑ 3.7)
	10	29.4$_{\pm 3.0}$	61.4$_{\pm 0.4}$	64.0$_{\pm 0.1}$	67.1$_{\pm 0.2}$ (↑ 3.1)	68.5$_{\pm 0.4}$	**69.4**$_{\pm 0.8}$ (↑ 0.9)	32.5$_{\pm 1.1}$	52.6$_{\pm 2.8}$	60.9$_{\pm 0.5}$	62.4$_{\pm 0.4}$ (↑ 1.5)	66.3$_{\pm 0.8}$	**67.6**$_{\pm 0.5}$ (↑ 1.3)
	50	40.9$_{\pm 0.3}$	80.4$_{\pm 0.4}$	82.8$_{\pm 0.6}$	84.0$_{\pm 0.3}$ (↑ 1.2)	84.6$_{\pm 0.7}$	**85.1**$_{\pm 0.3}$ (↑ 0.5)	60.8$_{\pm 1.0}$	74.5$_{\pm 2.5}$	81.6$_{\pm 0.4}$	82.3$_{\pm 0.3}$ (↑ 0.7)	85.4$_{\pm 0.3}$	**85.9**$_{\pm 0.2}$ (↑ 0.5)
	Whole dataset				90.2						89.1		
ImageWoof	1	13.3$_{\pm 0.5}$	**20.8**$_{\pm 1.2}$	19.3$_{\pm 0.3}$	20.2$_{\pm 0.7}$ (↑ 0.9)	19.4$_{\pm 0.5}$	14.9$_{\pm 0.9}$ (↓ 4.5)	12.2$_{\pm 2.2}$	21.1$_{\pm 1.9}$	**21.6**$_{\pm 0.5}$	16.5$_{\pm 0.7}$ (↓ 5.1)	18.4$_{\pm 0.2}$	15.1$_{\pm 0.7}$ (↓ 3.3)
	10	20.2$_{\pm 0.2}$	38.5$_{\pm 2.1}$	43.8$_{\pm 0.2}$	44.5$_{\pm 0.5}$ (↑ 0.7)	47.2$_{\pm 0.2}$	**48.3**$_{\pm 0.6}$ (↑ 1.1)	19.8$_{\pm 1.4}$	40.9$_{\pm 3.1}$	37.8$_{\pm 0.6}$	34.0$_{\pm 1.0}$ (↓ 3.8)	**44.9**$_{\pm 0.9}$	43.5$_{\pm 1.2}$ (↓ 1.4)
	50	23.3$_{\pm 0.3}$	68.5$_{\pm 0.7}$	67.7$_{\pm 1.0}$	69.6$_{\pm 0.9}$ (↑ 1.9)	73.0$_{\pm 0.6}$	**74.4**$_{\pm 0.5}$ (↑ 1.4)	35.7$_{\pm 0.4}$	59.8$_{\pm 3.2}$	66.5$_{\pm 1.1}$	68.0$_{\pm 1.0}$ (↑ 1.5)	72.7$_{\pm 0.7}$	**73.7**$_{\pm 0.7}$ (↑ 1.0)
	Whole dataset				85						80.3		
ImageNet-1K	10	21.3$_{\pm 0.6}$	42.0$_{\pm 0.1}$	47.8$_{\pm 0.4}$	48.8$_{\pm 0.2}$ (↑ 1.0)	50.8$_{\pm 0.1}$	**51.6**$_{\pm 0.3}$ (↑ 0.8)	28.4$_{\pm 0.1}$	–	45.6$_{\pm 1.6}$	45.3$_{\pm 0.8}$ (↓ 0.3)	56.9$_{\pm 0.2}$	**57.1**$_{\pm 0.5}$ (↑ 0.2)
	50	46.8$_{\pm 0.2}$	56.5$_{\pm 0.1}$	60.9$_{\pm 0.1}$	**61.3**$_{\pm 0.1}$ (↑ 0.4)	59.2$_{\pm 0.2}$	59.8$_{\pm 0.1}$ (↑ 0.6)	55.6$_{\pm 0.3}$	–	65.0$_{\pm 0.1}$	65.6$_{\pm 0.2}$ (↑ 0.6)	66.4$_{\pm 0.3}$	**66.6**$_{\pm 0.1}$ (↑ 0.2)
	Whole dataset				69.8						76.1		



(a) (b)

Figure 3: With ResNet18 as the teacher and IPC $= 50$, the difficulty distributions of the original data and the data distilled by FADRM and PSM on (a) CIFAR-10/100. (b) Runtime and peak GPU memory usage of GPS, as well as SUA and distillation for a single id_{IPC}, with IPC $= 10$. "//" indicates that the corresponding value exceeds the plotting range.

tions, enabling students to learn sample patterns ranging from easy to hard more comprehensively, and validating the robustness and effectiveness of PSM.

Efficiency. To evaluate the additional overhead of GPS computation and SUA forward passes, we measure their runtime and peak GPU memory. Since PSM and FADRM use the same distillation process, their efficiency difference mainly arises from these two steps. The costs of SUA and distillation are measured for the distilled batch corresponding to a single id_{IPC}. As shown in Figure 3b, GPS, SUA, and distillation require similar amounts of memory on small datasets, while distillation requires substantially more memory on Tiny-ImageNet. GPS and SUA also require much less time than distillation because they involve no gradient computation. Therefore, PSM introduces limited additional overhead while maintaining high computational efficiency and improving performance.

5.2.3 Cross-architecture Generalization.

Small-scale datasets. We evaluate cross-architecture generalization on CIFAR-10/100 and ImageNette/Woof using students with different scales and architectures. As shown in Table 2, under

Table 2: Cross-architecture Top-1 accuracy on small-scale datasets with IPC $= 10$. FADRM and PSM use ResNet18 as the teacher. Results evaluated with ResNet18 as the student are also reported.

Datasets		CIFAR-10		CIFAR-100		ImageNette		ImageWoof	
Student	Parameters	FADRM	PSM	FADRM	PSM	FADRM	PSM	FADRM	PSM
ShuffleNetV2	2.3M	$39.3_{\pm 1.0}$	$41.5_{\pm 0.8}$ (↑ 2.2)	$57.2_{\pm 0.3}$	$58.6_{\pm 0.3}$ (↑ 1.4)	$48.4_{\pm 1.1}$	$52.1_{\pm 0.2}$ (↑ 3.7)	$30.5_{\pm 2.2}$	$28.9_{\pm 1.6}$ (↓ 1.6)
MobileNetV2	3.4M	$43.0_{\pm 0.4}$	$41.3_{\pm 0.5}$ (↓ 1.7)	$56.4_{\pm 0.2}$	$57.8_{\pm 0.3}$ (↑ 1.4)	$54.2_{\pm 0.2}$	$56.1_{\pm 0.6}$ (↑ 1.9)	$30.1_{\pm 1.0}$	$32.0_{\pm 1.4}$ (↑ 1.9)
DenseNet121	8.0M	$48.0_{\pm 0.6}$	$46.0_{\pm 1.0}$ (↓ 2.0)	$62.1_{\pm 0.3}$	$63.9_{\pm 0.2}$ (↑ 1.8)	$66.3_{\pm 0.6}$	$67.8_{\pm 1.0}$ (↑ 1.5)	$43.0_{\pm 1.0}$	$43.6_{\pm 0.5}$ (↑ 0.6)
ResNet18	11.7M	$48.2_{\pm 0.4}$	$54.5_{\pm 1.3}$ (↑ 6.3)	$64.7_{\pm 0.4}$	$66.7_{\pm 0.4}$ (↑ 2.0)	$64.0_{\pm 0.1}$	$67.1_{\pm 0.2}$ (↑ 3.1)	$43.8_{\pm 0.2}$	$44.5_{\pm 0.5}$ (↑ 0.7)
ResNet50	25.6M	$36.5_{\pm 1.0}$	$40.2_{\pm 1.3}$ (↑ 3.7)	$60.2_{\pm 0.2}$	$62.6_{\pm 0.7}$ (↑ 2.4)	$62.4_{\pm 0.7}$	$65.0_{\pm 1.0}$ (↑ 2.6)	$40.5_{\pm 0.4}$	$40.5_{\pm 1.1}$ (= 0.0)
ResNet101	44.5M	$39.8_{\pm 0.1}$	$38.4_{\pm 1.0}$ (↓ 1.4)	$59.3_{\pm 1.0}$	$61.7_{\pm 0.4}$ (↑ 2.4)	$55.7_{\pm 0.7}$	$62.2_{\pm 0.8}$ (↑ 6.5)	$36.9_{\pm 0.7}$	$37.2_{\pm 0.2}$ (↑ 0.3)

Datasets		Tiny-ImageNet		ImageNet-1K	
Student	Parameters	FADRM	PSM	FADRM	PSM
DeiT-Ti	5.7M	$18.6_{\pm 0.3}$	$19.9_{\pm 0.2}$ (↑ 1.3)	$36.5_{\pm 0.5}$	$37.3_{\pm 0.1}$ (↑ 0.8)
ResNet18	11.7M	$51.4_{\pm 0.1}$	$51.7_{\pm 0.1}$ (↑ 0.3)	$60.9_{\pm 0.1}$	$61.3_{\pm 0.1}$ (↑ 0.4)
ResNet101	44.5M	$51.1_{\pm 1.8}$	$53.1_{\pm 0.9}$ (↑ 2.0)	$65.5_{\pm 0.1}$	$65.2_{\pm 0.1}$ (↓ 0.3)

Table 3: Cross-architecture Top-1 accuracy on large-scale datasets with IPC $= 50$. To further evaluate the generalization of the distilled data across different models, we also use DeiT (Touvron et al., 2021), which adopts a Transformer (Vaswani et al., 2017) architecture, as a student.

Table 4: Top-1 accuracy of different difficulty calculation methods with IPC $= 10$ and ResNet18 as both the teacher and student.

Difficulty Calculation	CIFAR-10	CIFAR-100	ImageNette	ImageWoof
Forgetting score	$\underline{48.5}_{\pm 0.2}$	$65.3_{\pm 0.2}$	$\underline{65.6}_{\pm 0.3}$	$39.6_{\pm 1.1}$
Confidence score	$43.3_{\pm 0.5}$	$\underline{65.7}_{\pm 0.6}$	$64.0_{\pm 0.4}$	$42.1_{\pm 0.6}$
Logits	$43.2_{\pm 1.1}$	$\underline{65.7}_{\pm 0.3}$	$64.8_{\pm 0.9}$	$\underline{43.8}_{\pm 0.6}$
GPS	$\mathbf{54.5}_{\pm 1.3}$	$\mathbf{66.7}_{\pm 0.4}$	$\mathbf{67.1}_{\pm 0.2}$	$\mathbf{44.5}_{\pm 0.5}$

Table 5: Top-1 accuracy in the ablation study of SUA and ISS with IPC $= 10$ and ResNet18 as both the teacher and student.

SUA	ISS	CIFAR-10	CIFAR-100	ImageNette	ImageWoof
		$48.2_{\pm 0.4}$	$64.7_{\pm 0.4}$	$64.0_{\pm 0.1}$	$\underline{43.8}_{\pm 0.2}$
✓		$\underline{50.7}_{\pm 0.5}$	$65.2_{\pm 0.4}$	$\underline{65.9}_{\pm 0.9}$	$42.1_{\pm 1.3}$
	✓	$47.6_{\pm 0.1}$	$\underline{65.3}_{\pm 0.2}$	$65.8_{\pm 0.4}$	$40.0_{\pm 0.8}$
✓	✓	$\mathbf{54.5}_{\pm 1.3}$	$\mathbf{66.7}_{\pm 0.4}$	$\mathbf{67.1}_{\pm 0.2}$	$\mathbf{44.5}_{\pm 0.5}$

most settings, PSM consistently outperforms FADRM across all datasets and student architectures. For example, on ImageNette, PSM outperforms FADRM by 6.5% when using ResNet101 (44.5M parameters) as the student, demonstrating its strong cross-architecture generalization ability.

Large-scale datasets. We conduct cross-architecture generalization experiments on Tiny-ImageNet and ImageNet-1K, and further include DeiT-Ti (5.7M parameters), which adopts a Transformer architecture, to evaluate the generalization of PSM across different architectures. As shown in Table 3, PSM performs well with both ResNet architecture and the newer DeiT architecture, demonstrating strong cross-architecture generalization on large-scale datasets.

5.3 Ablation Study

Difficulty Calculation Methods. We compare GPS with existing difficulty calculation methods, including Forgetting Score (Toneva et al., 2019), Confidence Score (Swayamdipta et al., 2020), and Logits (Chen et al., 2025). As shown in Table 4, GPS achieves the highest Top-1 accuracy on all datasets, indicating that it effectively partitions the original data into groups with distinct difficulties, provides reliable difficulty groups for SUA, and validates the effectiveness of PSM.

Epoch Ratio in SUA and Screening Method in ISS. SUA updates the teacher BN statistics through forward passes. We multiply the epoch ratio $r_{\text{fwd}} \in \{0.1, 0.2, \ldots, 1.0\}$ by the pretraining epochs E_k^{pre} of teacher k to determine the number of SUA forward epochs. For ISS, we compare four screening methods within each difficulty group: Random samples randomly, while Front, Middle, and Back select images from the front, middle, and back of the difficulty ranking, respectively.

We systematically study r_{fwd} and ISS methods on CIFAR-10/100 and ImageNette/Woof. As shown in Figure 4, Back performs best on CIFAR-10/100 and ImageWoof, while Front performs best on ImageNette. Under the optimal methods, CIFAR-10/100 (54.5% / 66.7%) and ImageNette (67.1%) achieve the highest accuracy at $r_{\text{fwd}} = 0.9/1.0$, whereas ImageWoof (44.5%) performs best at $r_{\text{fwd}} = 0.4$. Too few forward passes may update statistics insufficiently, while too many may over-adapt them to the current difficulty group. Thus, the optimal r_{fwd} depends on the dataset distribution.

Ablation of Mechanisms. SUA and ISS control distilled data difficulty through supervision and initialization, respectively. As shown in Table 5, removing both reduces PSM to FADRM. Individually, they improve several datasets but hurt ImageWoof, likely due to mismatched initialization and

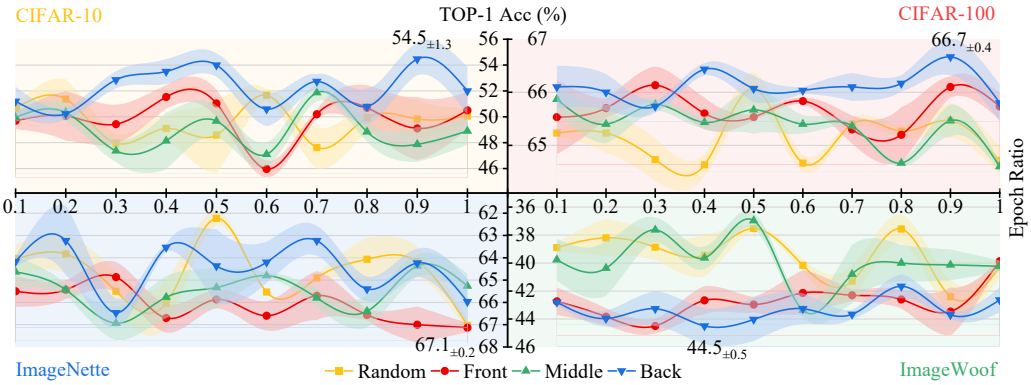


Figure 4: Hyperparameter study of SUA and ISS. On CIFAR-10/100 and ImageNette/Woof, we set IPC $= 10$ and use ResNet18 as both the teacher and student. We investigate the forward epoch ratio $r_{\text{fwd}} \in \{0.1, 0.2, \ldots, 1.0\}$ in SUA and the screening method (Random, Front, Middle, Back) in ISS.

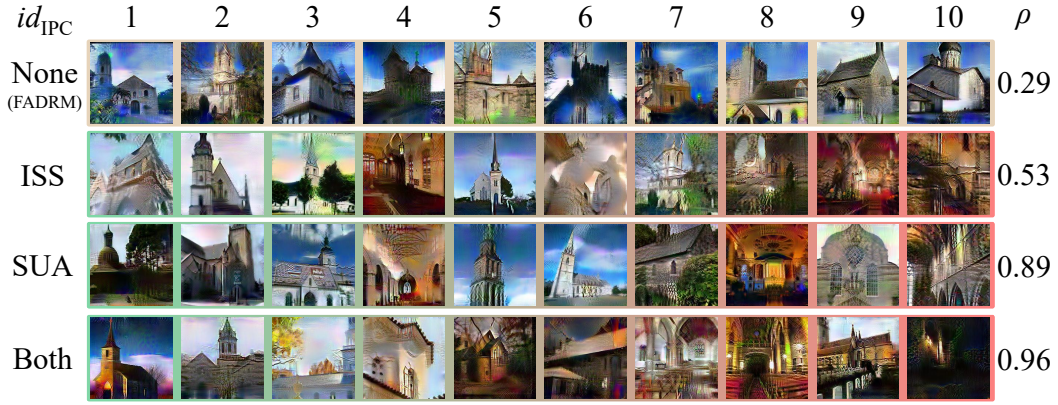


Figure 5: Distilled data on ImageNette with IPC $= 10$ and ResNet18 as the teacher, using neither mechanism (FADRM), ISS, SUA, or both. Spearman's ρ measures the agreement between the difficulty trends of the distilled and original data, with values closer to 1 indicating better agreement.

supervision. Jointly, they achieve the best results across all datasets, demonstrating their complementarity in constructing the difficulty structure.

During SUA, we use the same batch size as pretraining. For PSM+, all teachers share the same r_{fwd} and pretraining batch size. Further details and ablations are provided in the appendix C and D.

5.4 Visualization

Figure 5 visualizes the effect of PSM on distilled data difficulty. Spearman's ρ (Spearman, 1904) measures agreement between the difficulty trends of distilled and original data, with values closer to 1 indicating stronger agreement. FADRM produces similar data difficulties and the lowest ρ. Using ISS or SUA introduces clear difficulty variation, while combining both yields the clearest variation and the highest ρ, confirming that PSM effectively captures the difficulty structure.

6 Conclusion

In this work, we identify that existing decoupled methods struggle to capture the difficulty variation in the original data fully. To address this issue, we propose Precise Statistical Matching (PSM) by Difficulty. We first introduce GPS to rank and group original data by difficulty using BN statistics. We then propose SUA to update teacher BN statistics through efficient forward propagation, enabling difficulty-specific statistical supervision. Meanwhile, ISS selects real samples from the corresponding difficulty groups for initialization, providing distilled data with starting points consistent with the target difficulty. Extensive experiments show that PSM consistently improves student performance and better preserves the difficulty distribution of the original data. Future work will further improve PSM under extreme settings (e.g., IPC $= 1$), and investigate the effects of its parameters on datasets with different distributions (e.g., fine-grained datasets).

Ethics Statement

All datasets used in this work are publicly available benchmarks, including CIFAR-10/100, Tiny-ImageNet, and ImageNet-1K. They were obtained from publicly released sources and used in accordance with the corresponding terms of use. No new data involving human subjects were collected. This work focuses on DD based on sample difficulty modeling. Since the distillation process relies on the original data, the distilled data may preserve their distributional characteristics. Therefore, when applying distilled data to broader scenarios, their fairness and reliability should be evaluated.

Reproducibility Statement

Complete proofs of all propositions are provided in the appendix B.2. The pseudocode, distillation settings, additional baseline results, and ablation studies are also detailed in the appendix C and D. The supplementary material includes code for teacher pretraining, GPS computation, distillation, soft label generation, and evaluation, together with the scripts used at different experimental stages for each dataset. All experimental datasets are publicly available benchmarks. The complete code, pretrained teachers, and GPS results are included in the supplementary material and will be publicly released upon publication.

AI Use Statement

We did not use generative AI tools for any tasks requiring disclosure under the ICLR 2027 AI Policy for Authors. Large language models were used solely for language editing to improve the grammar and readability of the manuscript; they were not used to develop research ideas, prove mathematical claims, design experiments, analyze or interpret experimental results, generate figures, or conduct the literature review. The authors reviewed all AI-assisted edits to ensure that they preserved the intended technical meaning and did not alter the scientific claims, experimental results, or citations. We take full responsibility for the final content of this paper.

References

Wenqi Cai, Yawen Zou, Guang Li, Chunzhi Gu, and Chao Zhang. Evlf: Early vision-language fusion for generative dataset distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 33953–33962, 2026.

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)*, pp. 10718–10727, 2022.

Mingyang Chen, Bo Huang, Junda Lu, Bing Li, Yi Wang, Minhao Cheng, and Wei Wang. Dataset distillation via adversarial prediction matching. *arXiv preprint arXiv:2312.08912*, 2023.

Yanda Chen, Gongwei Chen, Miao Zhang, Weili Guan, and Liqiang Nie. Curriculum coarse-to-fine selection for high-ipc dataset distillation. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pp. 20437–20446, 2025.

Yutian Chen, Max Welling, and Alex Smola. Super-samples from kernel herding. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 2010.

Jiacheng Cui, Xinyue Bi, Yaxin Luo, Xiaohan Zhao, Jiacheng Liu, and Zhiqiang Shen. FADRM: Fast and accurate data residual matching for dataset distillation. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2025a.

Jiacheng Cui, Zhaoyi Li, Xiaochen Ma, Xinyue Bi, Yaxin Luo, and Zhiqiang Shen. Dataset distillation via committee voting. *arXiv preprint arXiv:2501.07575*, 2025b.

Jiacheng Cui, Bingkui Tong, Xinyue Bi, Xiaohan Zhao, Jiacheng Liu, and Zhiqiang Shen. Hard labels in! rethinking the role of hard labels in mitigating local semantic drift. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2026.

Justin Cui, Ruochen Wang, Si Si, and Cho-Jui Hsieh. Scaling up dataset distillation to imagenet-1k with constant memory. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 6565–6590, 2023.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255, 2009.

Ziang Gan, Qi Zhu, and Libao Zhang. Set-coupled guidance: Set-level coordination in diffusion-based dataset distillation. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2026.

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 (CVPR)*, pp. 15793–15803, 2024.

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.

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 (CVPR)*, pp. 770–778, 2016.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4700–4708, 2017.

Sergey Ioffe. Batch renormalization: Towards reducing minibatch dependence in batch-normalized models. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 448–456. PMLR, 2015.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

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

Saehyung Lee, Sanghyuk Chun, Sangwon Jung, Sangdoo Yun, and Sungroh Yoon. Dataset condensation with contrastive signals. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 12352–12364, 2022.

Yongmin Lee and Hye Won Chung. SelMatch: Effectively scaling up dataset distillation via selection-based initialization and partial updates by trajectory matching. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2024.

Shiye Lei and Dacheng Tao. A comprehensive survey of dataset distillation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(1):17–32, 2024.

Wenyuan Li, Guang Li, Keisuke Maeda, Takahiro Ogawa, and Miki Haseyama. Decoupled audio-visual dataset distillation. *arXiv preprint arXiv:2511.17890*, 2025.

Xuhui Li, Zhengquan Luo, Zihui Cui, and Zhiqiang Xu. Geodm: Geometry-aware distribution matching for dataset distillation. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2026.

Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting batch normalization for practical domain adaptation. In *International Conference on Learning Representations (ICLR) Workshop*, 2017.

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

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 (ICCV)*, pp. 17314–17324, 2023.

Ping Luo, Xinjiang Wang, Wenqi Shao, and Zhanglin Peng. Towards understanding regularization in batch normalization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

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)*, pp. 116–131, 2018.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205, 2023.

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 (ICCV)*, pp. 17097–17107, 2023.

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 (CVPR)*, pp. 4510–4520, 2018.

Mattia Sangermano, Antonio Carta, Andrea Cossu, and Davide Bacciu. Sample condensation in online continual learning. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2022.

Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How does batch normalization help optimization? In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Xinyi Shang, Peng Sun, Bei Shi, Zixuan Wang, and Tao Lin. Condensing large-scale datasets directly with minimal information loss. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2026.

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)*, pp. 16709–16718, 2024a.

Shitong Shao, Zikai Zhou, Huanran Chen, and Zhiqiang Shen. Elucidating the design space of dataset condensation. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2024b.

Guanghui Shi, Xuefeng Liang, and Qixiang Wen. Balanced dataset distillation via modeling multiple visual pattern distribution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19634–19643, 2026.

Charles Spearman. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101, 1904. doi: 10.2307/1412159.

Duo Su, Junjie Hou, Weizhi Gao, Yingjie Tian, and Bowen Tang. D4M: Dataset distillation via disentangled diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5809–5818, 2024.

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)*, pp. 9390–9399, 2024.

Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A. Smith, and Yejin Choi. Dataset cartography: Mapping and diagnosing datasets with training dynamics. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9275–9293, 2020.

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

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 10347–10357, 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

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)*, pp. 12196–12205, 2022.

Lingao Xiao and Yang He. Are large-scale soft labels necessary for large-scale dataset distillation? In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Eric Xue, Yijiang Li, Haoyang Liu, Peiran Wang, Yifan Shen, and Haohan Wang. Towards adversarially robust dataset distillation by curvature regularization. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2025.

Linfeng Ye, Shayan Mohajer Hamidi, Guang Li, Takahiro Ogawa, Miki Haseyama, and Konstantinos N. Plataniotis. Information-guided diffusion sampling for dataset distillation. In *Advances in Neural Information Processing Systems (NeurIPS), Workshop*, 2025.

Zeyuan Yin and Zhiqiang Shen. Dataset distillation in large data era. *Transactions on Machine Learning Research*, 2024.

Zeyuan Yin, Eric Xing, and Zhiqiang Shen. Squeeze, recover and relabel: Dataset condensation at imagenet scale from a new perspective. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

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

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

Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. Dataset condensation with gradient matching. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

Wenliang Zhong, Haoyu Tang, Qinghai Zheng, Mingzhu Xu, Yupeng Hu, and Weili Guan. Towards stable and storage-efficient dataset distillation: Matching convexified trajectory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025a.

Xinhao Zhong, Hao Fang, Bin Chen, Xulin Gu, Meikang Qiu, Shuhan Qi, and Shu-Tao Xia. Hierarchical features matter: A deep exploration of progressive parameterization method for dataset distillation. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pp. 30462–30471, 2025b.

A Additional Related Work

Dataset Distillation. Dataset distillation (DD) compresses a large dataset into a small set of samples with high training utility, substantially reducing storage and training costs while maintaining performance comparable to training on the full dataset. Existing methods mainly include gradient matching (Zhao et al., 2021; Zhao & Bilen, 2021; Lee et al., 2022; Liu et al., 2023), distribution matching (Zhao & Bilen, 2023; Wang et al., 2022; Sajedi et al., 2023; Xue et al., 2025; Li et al., 2025), trajectory matching (Cui et al., 2023; Chen et al., 2023; Guo et al., 2024), decoupled distillation (Yin et al., 2023; Cui et al., 2025a; Shao et al., 2024b; Yin & Shen, 2024; Xiao & He, 2024), and generative distillation (Zhong et al., 2025b; Gu et al., 2024; Su et al., 2024; Ye et al., 2025; Cai et al., 2026).

Decoupled Dataset Distillation. SRe^{2}L (Yin et al., 2023) pioneered decoupled DD by separating teacher pretraining from distilled data optimization. SRe^{2}L++ (Cui et al., 2025b) improves robustness through stronger data augmentation and soft labels specific to each batch. FADRM (Cui et al., 2025a) uses multiscale data residual connections to preserve original information and enrich distilled samples, while FADRM+ extends it to multiple teachers. G-VBSM (Shao et al., 2024a) employs lightweight model ensembles to improve generalization across architectures, while CDA (Yin & Shen, 2024) stabilizes optimization through a curriculum strategy. LPLD (Xiao & He, 2024) reexamines the need for large volumes of soft labels and provides a lighter alternative. Despite their strong performance on standard benchmarks, these methods do not explicitly consider the difficulty structure of the original data and its influence on distilled samples.

Batch Normalization. Batch Normalization (BN) (Ioffe & Szegedy, 2015) normalizes intermediate features using batch means and variances, improving training speed and stability. Batch Renormalization (Ioffe, 2017) uses running statistics to reduce dependence on batch composition and improve training with small batches. Santurkar et al. (Santurkar et al., 2018) show that BN smooths the optimization landscape and stabilizes gradients. Luo et al. (Luo et al., 2019) interpret BN as implicit regularization and analyze its effects on convergence and generalization. AdaBN (Li et al., 2017) adapts BN statistics across domains, showing that they characterize domain distributions. Overall, BN improves training stability, while its running means and variances capture layer-wise feature distributions, supporting their use as statistical supervision in decoupled dataset distillation.

B Theoretical Analysis

B.1 The Empirical Relationship Between GPS and Classification Difficulty

GPS measures the distance between the statistics produced by a sample at the inputs to the teacher's BN layers and the corresponding BN running statistics, thereby characterizing its deviation from the feature distribution learned by the teacher. To analyze the correlation between statistical deviation and classification difficulty, we measure sample difficulty using the teacher's cross-entropy loss on the ground truth label. For a sample (x_i, y_i), its classification difficulty is defined as

$$\ell_i = \mathcal{L}_{\mathrm{CE}}(f_\theta(x_i), y_i) = -\log p_\theta(y_i \mid x_i), \tag{8}$$

where $p_\theta(y_i \mid x_i)$ denotes the probability that the teacher assigns to the ground truth class y_i. A larger ℓ_i indicates lower confidence in the ground truth class and thus greater classification difficulty.

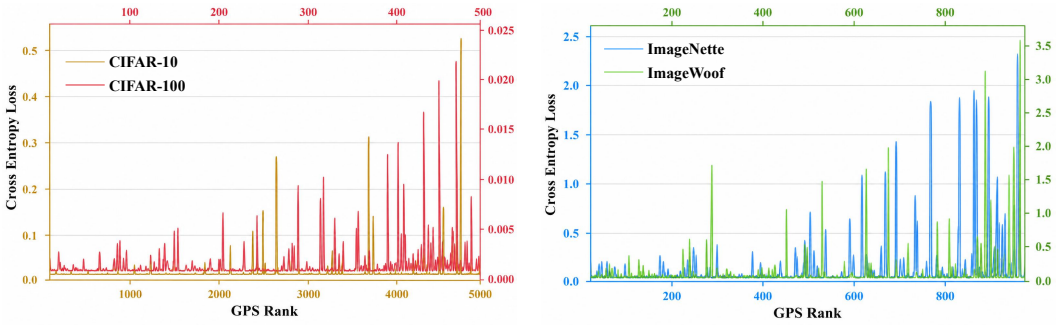
We conduct this analysis on ImageNette/Woof and CIFAR-10/100 using a pretrained ResNet18 teacher to compute the GPS of each original sample. Suppose the dataset contains N samples, and let π denote the index sequence obtained by sorting them in ascending order of GPS:

$$\mathrm{GPS}\big(x_{\pi(1)}\big) \le \mathrm{GPS}\big(x_{\pi(2)}\big) \le \cdots \le \mathrm{GPS}\big(x_{\pi(N)}\big)\,. \tag{9}$$

We then compute the corresponding cross-entropy loss $\ell_{\pi(r)}$ for each ordered sample and examine how the classification loss changes along the GPS ranking. As shown in Figure 6, the cross-entropy loss generally increases with GPS, and samples with larger GPS values tend to exhibit higher classification losses. This empirical result indicates that the statistical deviation measured by GPS varies consistently with the classification difficulty perceived by the teacher, supporting the use of GPS to rank and group the original samples from easy to difficult.

B.2 The Expected Convergence of Statistics

Proof of Proposition 1. Fix an arbitrary teacher k and difficulty group g. Let $\mathcal{B}_t$ denote the t-th forward batch independently drawn from $\mathcal{T}_g$, and let $\widehat{\tau}_{k,g}^{(t)}$ denote the batch statistics induced by $\mathcal{B}_t$



(a) CIFAR-10 and CIFAR-100. (b) ImageNette and ImageWoof.

Figure 6: Relationship between GPS and classification difficulty on different datasets.

at all BN layers. Since the teacher parameters are fixed and the forward batches are identically distributed, we have

$$\mathbb{E}\left[\widehat{\boldsymbol{\tau}}_{k,g}^{(t)}\right] = \boldsymbol{\tau}_{k,g}^{\star}, \qquad \forall t. \tag{10}$$

Applying the BN update rule in equation 3 componentwise to all running means and variances gives

$$\boldsymbol{\tau}_{k,g}^{(t)} = (1-\rho)\boldsymbol{\tau}_{k,g}^{(t-1)} + \rho\widehat{\boldsymbol{\tau}}_{k,g}^{(t)}, \tag{11}$$

where SUA starts from the global statistics stored in the pretrained teacher:

$$\boldsymbol{\tau}_{k,g}^{(0)} = \boldsymbol{\tau}_{k}^{\mathrm{G}}. \tag{12}$$

Taking expectations in equation 11 and using equation 10, we obtain

$$\mathbb{E}\left[\boldsymbol{\tau}_{k,g}^{(t)}\right] - \boldsymbol{\tau}_{k,g}^{\star} = (1-\rho)\left(\mathbb{E}\left[\boldsymbol{\tau}_{k,g}^{(t-1)}\right] - \boldsymbol{\tau}_{k,g}^{\star}\right). \tag{13}$$

Recursively applying equation 13 and substituting equation 12 yields

$$\mathbb{E}\left[\boldsymbol{\tau}_{k,g}^{(t)}\right] - \boldsymbol{\tau}_{k,g}^{\star} = (1-\rho)^{t}\left(\boldsymbol{\tau}_{k}^{\mathrm{G}} - \boldsymbol{\tau}_{k,g}^{\star}\right). \tag{14}$$

Since $\rho \in (0, 1]$, we have $0 \leq 1-\rho < 1$. Therefore,

$$\mathbb{E}\left[\boldsymbol{\tau}_{k,g}^{(t)}\right] - \boldsymbol{\tau}_{k,g}^{\star} \xrightarrow{t\to\infty} \mathbf{0}. \tag{15}$$

Thus, the expected discrepancy between the SUA running statistics and the statistics of the current difficulty group decreases geometrically with the number of forward batch updates. This result holds for the running means and variances of all BN layers. Consequently, SUA statistics converge in expectation to those of the current difficulty group and provide the corresponding statistical matching signal for distillation batch g. □

C More Results

C.1 Comparison with other SOTA Methods

To further evaluate the effectiveness, we compare PSM with more representative SOTA dataset distillation methods, including the coreset selection methods Random and Herding (Chen et al., 2010), the gradient matching method DSA (Zhao & Bilen, 2021), the distribution matching method DM (Zhao & Bilen, 2023), the trajectory matching methods MTT (Cazenavette et al., 2022), DATM (Guo et al., 2024) and TESLA (Cui et al., 2023), the generative method Minimax (Gu et al., 2024), and the decoupled methods SRe^{2}L and FADRM. Specifically, DSA uses ConvNet as the teacher model, DM uses ConvNet and ResNetAP-10, MTT uses ConvNet-BN, DATM and TESLA use ConvNet-IN, and Minimax uses DiT (Peebles & Xie, 2023), while SRe^{2}L, FADRM, and PSM use ResNet18. All methods are evaluated using ResNet18 as the student model.

As shown in Table 6, coreset selection methods generally achieve lower accuracy, but gradually approach DD methods as IPC increases. For example, Random achieves 75.8% on ImageNette with IPC = 50. DD methods perform better in most settings, where gradient matching, distribution

Table 6: Top-1 accuracy (%) comparison of representative dataset distillation methods with ResNet18 as the student. Standard deviations are reported when available.

Dataset	IPC	Coreset		Gradient	Distribution	Trajectory			Generative	Decoupled		
		Random	Herding	DSA	DM	MTT	DATM	TESLA	Minimax	SRe^{2}L	FADRM	PSM
CIFAR-10	1	–	–	–	–	–	–	–	–	$16.6_{\pm 0.9}$	$\underline{19.3}_{\pm 0.6}$	$\mathbf{20.9}_{\pm 0.1}$
	10	$25.1_{\pm 0.5}$	$28.4_{\pm 0.1}$	$42.1_{\pm 0.6}$	$38.2_{\pm 1.1}$	$46.1_{\pm 1.4}$	48.66	$\underline{48.9}_{\pm 2.2}$	–	$29.3_{\pm 0.5}$	$48.2_{\pm 0.4}$	$\mathbf{54.5}_{\pm 1.3}$
	50	54.96	–	$47.8_{\pm 0.9}$	$52.9_{\pm 0.4}$	$58.7_{\pm 0.2}$	66.27	–	–	$45.0_{\pm 0.7}$	$\underline{80.6}_{\pm 0.7}$	$\mathbf{81.1}_{\pm 0.1}$
CIFAR-100	1	–	–	–	–	–	–	–	–	$6.6_{\pm 0.2}$	$\underline{31.3}_{\pm 0.2}$	$\mathbf{32.9}_{\pm 0.9}$
	10	$10.9_{\pm 0.1}$	$13.3_{\pm 0.3}$	$21.9_{\pm 0.4}$	$18.7_{\pm 0.5}$	$26.8_{\pm 0.6}$	–	$27.1_{\pm 0.7}$	–	$27.0_{\pm 0.4}$	$\underline{64.7}_{\pm 0.4}$	$\mathbf{66.7}_{\pm 0.4}$
	50	$40.7_{\pm 1.0}$	–	$43.6_{\pm 0.7}$	$42.6_{\pm 0.5}$	$51.3_{\pm 0.4}$	$51.0_{\pm 0.5}$	–	–	$50.2_{\pm 0.4}$	$\underline{69.3}_{\pm 0.3}$	$\mathbf{69.7}_{\pm 0.3}$
ImageNette	1	–	–	–	–	–	–	–	–	$19.1_{\pm 1.1}$	$\mathbf{28.2}_{\pm 0.6}$	$\underline{25.0}_{\pm 0.5}$
	10	$55.8_{\pm 1.0}$	–	–	$60.9_{\pm 0.7}$	–	–	–	$\underline{64.9}_{\pm 0.6}$	$29.4_{\pm 3.0}$	$64.0_{\pm 0.1}$	$\mathbf{67.1}_{\pm 0.2}$
	50	$75.8_{\pm 1.1}$	–	–	$75.0_{\pm 1.0}$	–	–	–	$78.1_{\pm 0.6}$	$40.9_{\pm 0.3}$	$\underline{82.8}_{\pm 0.6}$	$\mathbf{84.0}_{\pm 0.3}$
ImageWoof	1	–	–	–	–	–	–	–	–	$13.3_{\pm 0.5}$	$\underline{19.3}_{\pm 0.3}$	$\mathbf{20.2}_{\pm 0.7}$
	10	$27.7_{\pm 0.9}$	$30.2_{\pm 1.2}$	–	$33.4_{\pm 0.7}$	–	–	–	$37.6_{\pm 0.9}$	$20.2_{\pm 0.2}$	$\underline{43.8}_{\pm 0.2}$	$\mathbf{44.5}_{\pm 0.5}$
	50	$47.9_{\pm 1.8}$	$48.3_{\pm 1.2}$	–	$46.2_{\pm 0.6}$	–	–	–	$57.1_{\pm 0.6}$	$23.3_{\pm 0.3}$	$\underline{67.7}_{\pm 1.0}$	$\mathbf{69.6}_{\pm 0.9}$
Tiny-ImageNet	1	–	–	–	–	–	–	–	–	$2.6_{\pm 0.1}$	$\underline{28.6}_{\pm 0.1}$	$\mathbf{30.6}_{\pm 1.3}$
	10	$7.5_{\pm 0.1}$	$9.0_{\pm 0.3}$	–	–	–	–	–	–	$16.1_{\pm 0.2}$	$\underline{46.5}_{\pm 0.4}$	$\mathbf{48.1}_{\pm 0.3}$
	50	$30.1_{\pm 0.6}$	–	$27.8_{\pm 1.4}$	$31.0_{\pm 0.6}$	$40.3_{\pm 0.3}$	$42.2_{\pm 0.2}$	–	–	$41.1_{\pm 0.4}$	$\underline{51.4}_{\pm 0.1}$	$\mathbf{51.7}_{\pm 0.1}$
ImageNet-1K	10	$4.4_{\pm 0.1}$	$5.8_{\pm 0.1}$	–	–	–	–	$7.7_{\pm 0.1}$	$44.3_{\pm 0.5}$	$21.3_{\pm 0.6}$	$\underline{47.8}_{\pm 0.4}$	$\mathbf{48.8}_{\pm 0.2}$
	50	–	–	–	–	–	–	–	$58.6_{\pm 0.3}$	$46.8_{\pm 0.2}$	$\underline{60.9}_{\pm 0.1}$	$\mathbf{61.3}_{\pm 0.1}$

Table 7: Top-1 accuracy under different epoch ratios on ImageNette with IPC $= 10$ under the same-architecture protocol.

Epoch Ratio	DenseNet121	MobileNetV2	ShuffleNetV2	ResNet18
0.1	$68.6_{\pm 0.6}$	$51.5_{\pm 0.6}$	$49.1_{\pm 1.1}$	$65.5_{\pm 0.7}$
0.2	$68.8_{\pm 0.7}$	$51.8_{\pm 1.0}$	$\underline{50.2}_{\pm 2.5}$	$65.4_{\pm 0.4}$
0.3	$68.9_{\pm 1.0}$	$50.0_{\pm 0.7}$	$48.8_{\pm 0.1}$	$64.9_{\pm 0.7}$
0.4	$68.3_{\pm 0.9}$	$53.7_{\pm 1.1}$	$48.0_{\pm 2.1}$	$66.7_{\pm 0.7}$
0.5	$67.4_{\pm 1.1}$	$51.1_{\pm 0.5}$	$\mathbf{50.3}_{\pm 1.3}$	$65.9_{\pm 0.4}$
0.6	$67.2_{\pm 0.2}$	$52.3_{\pm 0.8}$	$50.0_{\pm 1.5}$	$66.6_{\pm 0.7}$
0.7	$67.7_{\pm 1.2}$	$\mathbf{54.2}_{\pm 0.7}$	$50.1_{\pm 1.6}$	$65.7_{\pm 1.1}$
0.8	$68.9_{\pm 0.3}$	$52.8_{\pm 1.5}$	$49.1_{\pm 0.8}$	$66.6_{\pm 0.2}$
0.9	$\underline{69.1}_{\pm 0.6}$	$\underline{53.9}_{\pm 3.4}$	$49.4_{\pm 1.3}$	$\underline{67.0}_{\pm 0.8}$
1.0	$\mathbf{69.1}_{\pm 0.7}$	$52.0_{\pm 0.8}$	$49.5_{\pm 0.2}$	$\mathbf{67.1}_{\pm 0.2}$

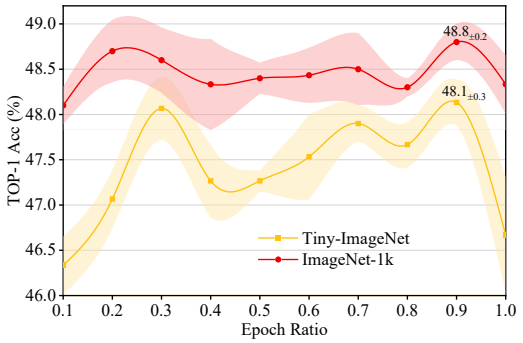


Figure 7: Top-1 accuracy under different epoch ratios on Tiny-ImageNet and ImageNet-1K with IPC $= 10$ and teacher=student=ResNet18.

matching, trajectory matching, and generative methods use gradients, feature distributions, weight trajectories, and generative priors to guide distillation, respectively. Although these methods compress data effectively, their supervision signals usually describe the overall data distribution in an aggregated form, making it difficult to capture the internal difficulty structure. As dataset scale increases, sample difficulty differences are more easily smoothed by aggregated objectives, causing distilled data to concentrate at similar difficulties and limiting coverage of the original difficulty range. Among decoupled methods, FADRM performs strongly, while PSM preserves a broader difficulty structure, enabling students to learn sample patterns ranging from easy to difficult, and achieving the best results in most settings.

C.2 More Ablation Study

Optimal Epoch Ratios for other Models. To determine the optimal epoch ratio r_{fwd} for the other teacher models (DenseNet121, MobileNetV2, and ShuffleNetV2) in PSM+, we conduct an ablation study on ImageNette under the same-architecture protocol with IPC $= 10$. As shown in Table 7, the optimal r_{fwd} values for DenseNet121, MobileNetV2, and ShuffleNetV2 are 1.0 (69.1%), 0.7 (54.2%), and 0.5 (50.3%), respectively. Overall, models with more parameters require more forward passes to better capture the distribution of the current difficulty group. Since the performance gaps

Table 8: Top-1 accuracy under different patch grid settings with IPC $= 10$, using ResNet18 as both the teacher and student models.

Patch Grid	CIFAR-10	CIFAR-100	ImageNette	ImageWoof
1×1	$54.5_{\pm 1.3}$	$66.7_{\pm 0.4}$	$67.1_{\pm 0.2}$	$44.5_{\pm 0.5}$
2×2	$54.3_{\pm 1.9}$	$65.0_{\pm 0.4}$	$65.1_{\pm 0.9}$	$40.2_{\pm 1.9}$



Figure 8: Visualization of different patch grids.

Table 9: Hyperparameter settings during pretraining and GPS.

Hyperparameter	CIFAR-10	CIFAR-100	Tiny-ImageNet	ImageNette	ImageWoof	ImageNet-1K
learning rate	1.00E−03	1.00E−03	1.00E−02	1.00E−02	1.00E−02	
optimizer	Adam	Adam	SGD	SGD	SGD	PyTorch
scheduler	cos	cos	cos	cos	cos	
epoch	150	100	50	150	200	90
batch	512	512	128	64	64	64
β	1	1	1	1	1	1
ϵ	1.00E−06	1.00E−06	1.00E−06	1.00E−06	1.00E−06	1.00E−06

between the optimal r_{fwd} and $r_{\text{fwd}} = 1.0$ are small for MobileNetV2 and ShuffleNetV2, we apply the optimal r_{fwd} determined for ResNet18 to the other teacher models in PSM+ on other datasets and IPC settings for simplicity.

Epoch Ratio r_{fwd} on Tiny-ImageNet and ImageNet-1K. Due to limited computational resources, we do not further investigate the effect of the ISS screening methods on Tiny-ImageNet and ImageNet-1K. Instead, we only examine the effect of different r_{fwd} under IPC $= 10$, using ResNet18 as the teacher and student. As shown in Figure 7, consistent with the observations in Figure 4, larger r_{fwd} allows the statistics to better characterize the distribution of the current difficulty group. Both Tiny-ImageNet and ImageNet-1K achieve the highest accuracy at $r_{\text{fwd}} = 0.9$, reaching 48.1% and 48.8%, respectively. Therefore, we set $r_{\text{fwd}} = 0.9$ for the other experiments on Tiny-ImageNet and ImageNet-1K.

Patch Grids in the Initial Image. Since PSM modifies the sample initialization method, we further investigate the effect of different patch grids on performance. We use ResNet18 as the teacher and student with IPC $= 10$, and the specific patch grid settings are illustrated in Figure 8. As shown in Table 8, the 1×1 patch grid achieves the best performance on all datasets, consistent with the findings of (Cui et al., 2025a). When the patch grid increases to 2×2, compressing local image regions may cause information loss, which is more evident on high-resolution datasets. Therefore, we use the 1×1 initialization setting in all experiments.

Algorithm 1 Global Precision Score (GPS)

Require: Original set $\mathcal{T} = \{(x_i, y_i)\}_{i=1}^{N}$; pretrained teachers $\{f_{\theta_k}\}_{k=1}^{K}$; repeated evaluations R

for each teacher $k = 1, \ldots, K$ **do**
  for $r = 1, \ldots, R$ **do**
    for each sample $x_i \in \mathcal{T}$ **do**
      Forward x_i through f_{θ_k} and compute its statistics at all BN layers
      Compute $s_{i,k}^{(r)}$ using the statistical distances in equation 4
    end for
  end for
  $s_{i,k} \leftarrow \frac{1}{R} \sum_{r=1}^{R} s_{i,k}^{(r)}$
  Rank $s_{i,k}$ within each class: $r_{i,k} \leftarrow \text{rank}_c(s_{i,k})$
end for
$\text{GPS}(x_i) \leftarrow \frac{1}{K} \sum_{k=1}^{K} r_{i,k}$ for each $x_i \in \mathcal{T}$
return Class-wise GPS rankings from easy to hard

Table 10: Hyperparameter settings during distillation.

Hyperparameter	CIFAR-10	CIFAR-100	Tiny-ImageNet	ImageNette	ImageWoof	ImageNet-1K
Epoch ratio	0.9	0.9	0.9	1.0	0.4	0.9
Screening method	Back	Back	Back	Front	Back	Back
PSM+ teachers	ShuffleNetV2, MobileNetV2, DenseNet121, ResNet18					

Algorithm 2 Precise Statistical Matching

Require: Difficulty groups $\{\mathcal{T}_{g,c}\}$ obtained by GPS; pretrained teachers $\{f_{\theta_k}\}_{k=1}^{K}$; forward ratio r_{fwd}; pretraining epochs $\{E_k^{\mathrm{pre}}\}_{k=1}^{K}$; optimization iterations T

Ensure: Distilled dataset $\widetilde{\mathcal{T}}$

Initialize $\widetilde{\mathcal{T}} \leftarrow \varnothing$
for $g = 1, \ldots, \mathrm{IPC}$ **do**
  SUA: construct difficulty-specific statistical supervision
  for each teacher $k = 1, \ldots, K$ **do**
    Freeze θ_k and initialize BN statistics with $\boldsymbol{\tau}_k^{\mathrm{G}}$
    Forward $\mathcal{T}_g = \bigcup_c \mathcal{T}_{g,c}$ through f_{θ_k} for $\lfloor r_{\mathrm{fwd}} E_k^{\mathrm{pre}} \rfloor$ epochs
    Update only BN running statistics according to equation 3
    Store the updated statistics as $\boldsymbol{\tau}_{k,g}^{\mathrm{SUA}}$
  end for
  ISS: initialize distilled data with matched difficulty
  for each class c **do**
    Select real samples from $\mathcal{T}_{g,c}$ using ISS
    Initialize $\widetilde{x}_{g,c}^{(0)}$ with the selected samples
  end for
  Distillation:
  for $t = 1, \ldots, T$ **do**
    Select teacher k and use $\boldsymbol{\tau}_{k,g}^{\mathrm{SUA}}$ as its statistical target
    Update $\widetilde{\mathcal{T}}_g$ by minimizing equation 2 with the SUA statistics
  end for
  $\widetilde{\mathcal{T}} \leftarrow \widetilde{\mathcal{T}} \cup \widetilde{\mathcal{T}}_g$
end for
return $\widetilde{\mathcal{T}}$

D Implementation Details

D.1 Pretraining.

As shown in Table 9, we report the pretraining hyperparameters for ShuffleNetV2, ResNet18, ResNet50, MobileNetV2, and DenseNet121 on each dataset. To ensure that all teacher models in PSM+ undergo the same number of forward passes under the same r_{fwd}, we use the same number of pretraining epochs and the same pretraining configuration for all models on each dataset. For ImageNet-1K, we directly use the official pretrained weights provided by PyTorch.

After pretraining the teacher models, we compute GPS. The batch size is kept the same as in pretraining, with $\beta = 1.0$ and $\epsilon = 1 \times 10^{-6}$. The detailed procedure is shown in Algorithm 1.

D.2 Distillation.

For PSM+, to simplify the experimental setup, we use ShuffleNetV2, ResNet18, MobileNetV2, and DenseNet121 as teacher models on all datasets, while using the same teacher model settings as FADRM+. During the forward passes of SUA, the batch size fed into the teacher models is kept the same as that used during pretraining to reduce statistical bias. The remaining hyperparameters are shown in Table 10, and the detailed procedure is provided in Algorithm 2. It is worth noting that when $\mathrm{IPC} = 1$, there is only one difficulty group, meaning that the entire dataset is treated as a single group. In this case, only ISS takes effect in PSM. Therefore, as shown in Table 1, PSM and FADRM(+) achieve similar performance on most datasets when $\mathrm{IPC} = 1$.

D.3 Soft Label Generation.

To maintain consistency in the soft label distribution, we use the original pretrained teachers to generate soft labels, rather than the teachers after forward passes. For FADRM+ and PSM+, we assign equal weights to all teachers during soft label generation to simplify the experimental setup.

E More Visualization

To more intuitively illustrate the changes in the difficulty of distilled samples, we visualize the samples generated by FADRM and PSM with ResNet18 as the teacher model and $\mathrm{IPC} = 10$. Due to space limitations, we only show results on CIFAR-10, ImageNette, and ImageWoof.

As shown in Figure 9, 11 and 13, the distilled samples generated by FADRM are typically concentrated within a relatively narrow difficulty range. In contrast, as shown in Figure 10, 12 and 14, PSM generates distilled samples that cover a broader difficulty range, promoting the student model to learn a more complete difficulty structure.



Figure 9: Visualization of the distilled CIFAR-10 samples generated by FADRM with $\mathrm{IPC} = 10$ and teacher=ResNet18.

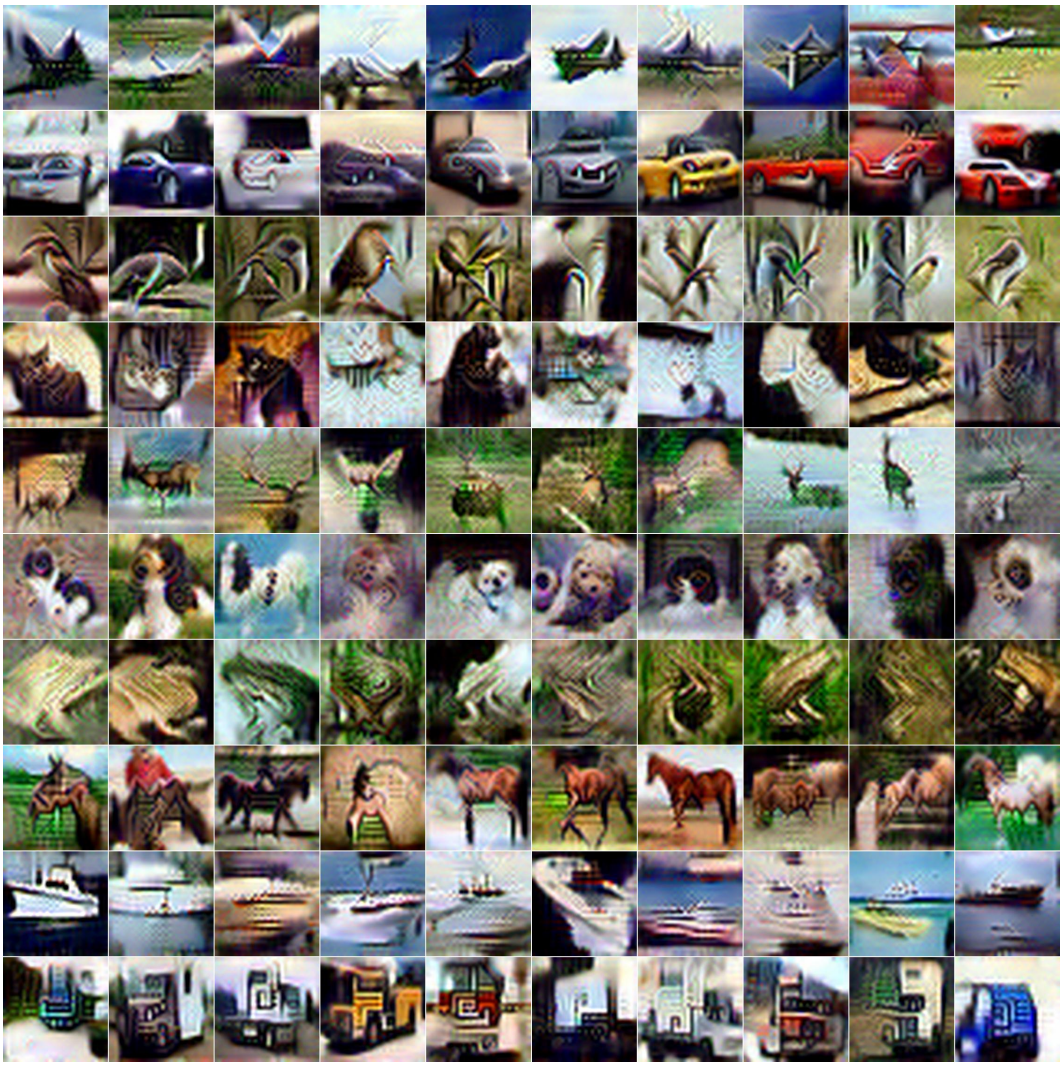


Figure 10: Visualization of the distilled CIFAR-10 samples generated by PSM with IPC $= 10$ and teacher=ResNet18.

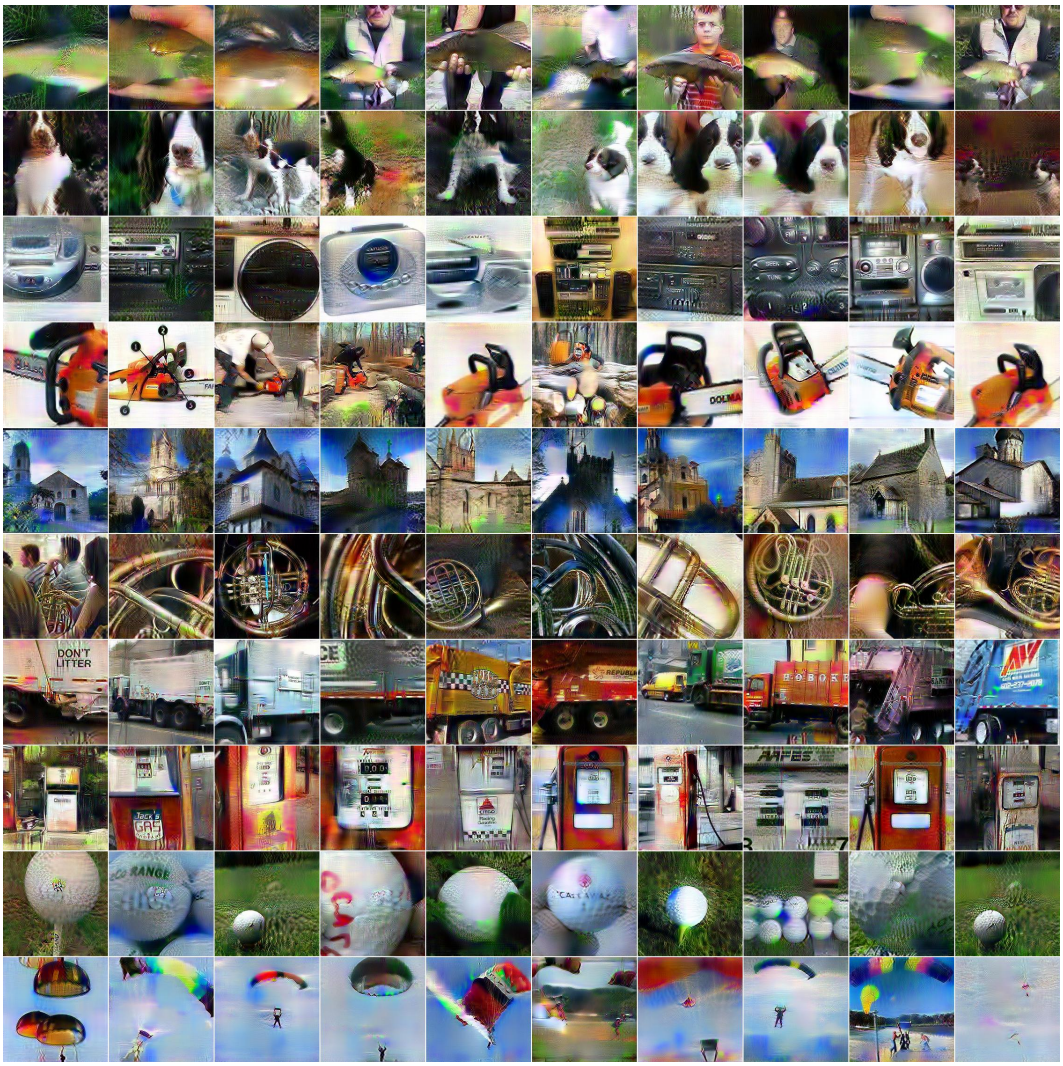


Figure 11: Visualization of the distilled ImageNette samples generated by FADRM with IPC $= 10$ and teacher=ResNet18.

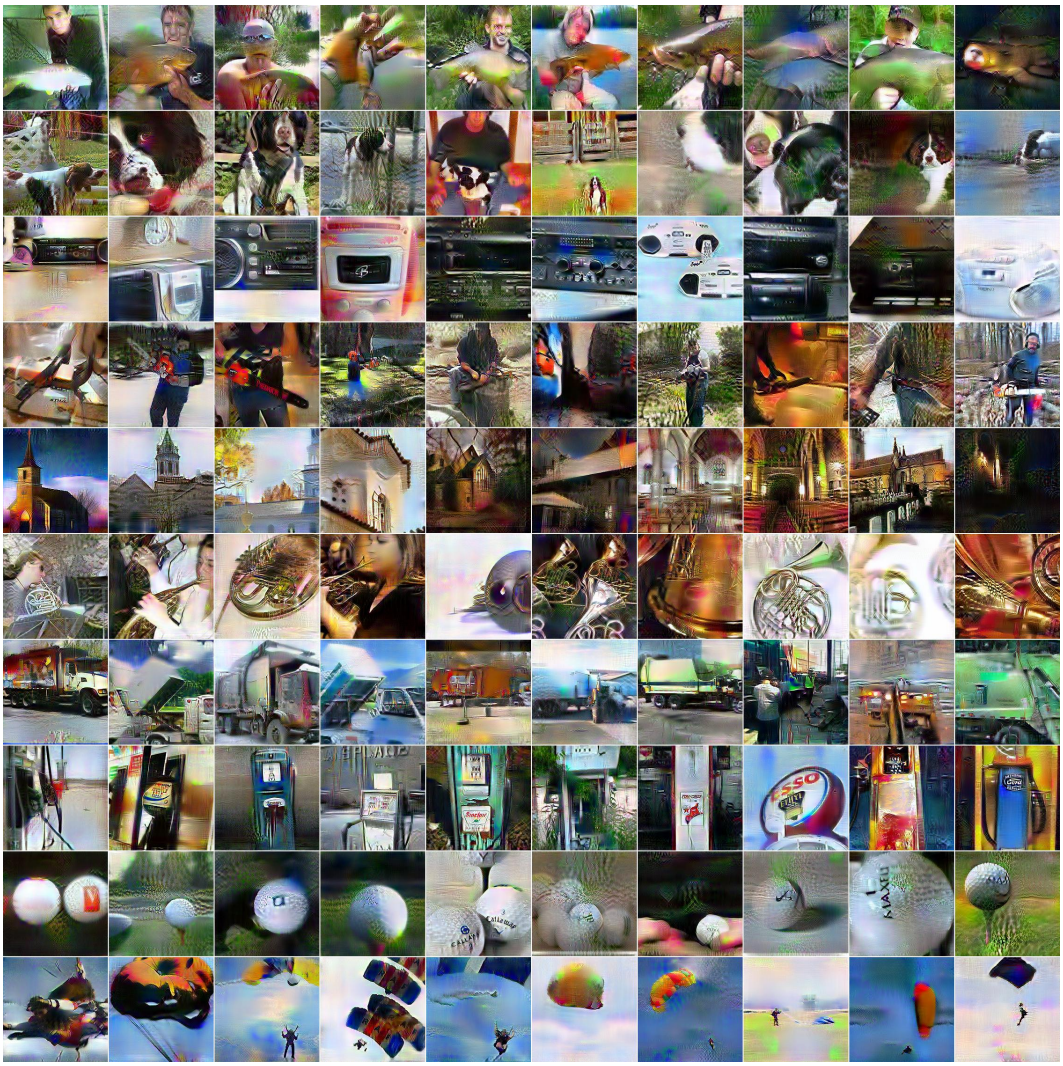


Figure 12: Visualization of the distilled ImageNette samples generated by PSM with IPC $= 10$ and teacher=ResNet18.



Figure 13: Visualization of the distilled ImageWoof samples generated by FADRM with IPC $= 10$ and teacher=ResNet18.

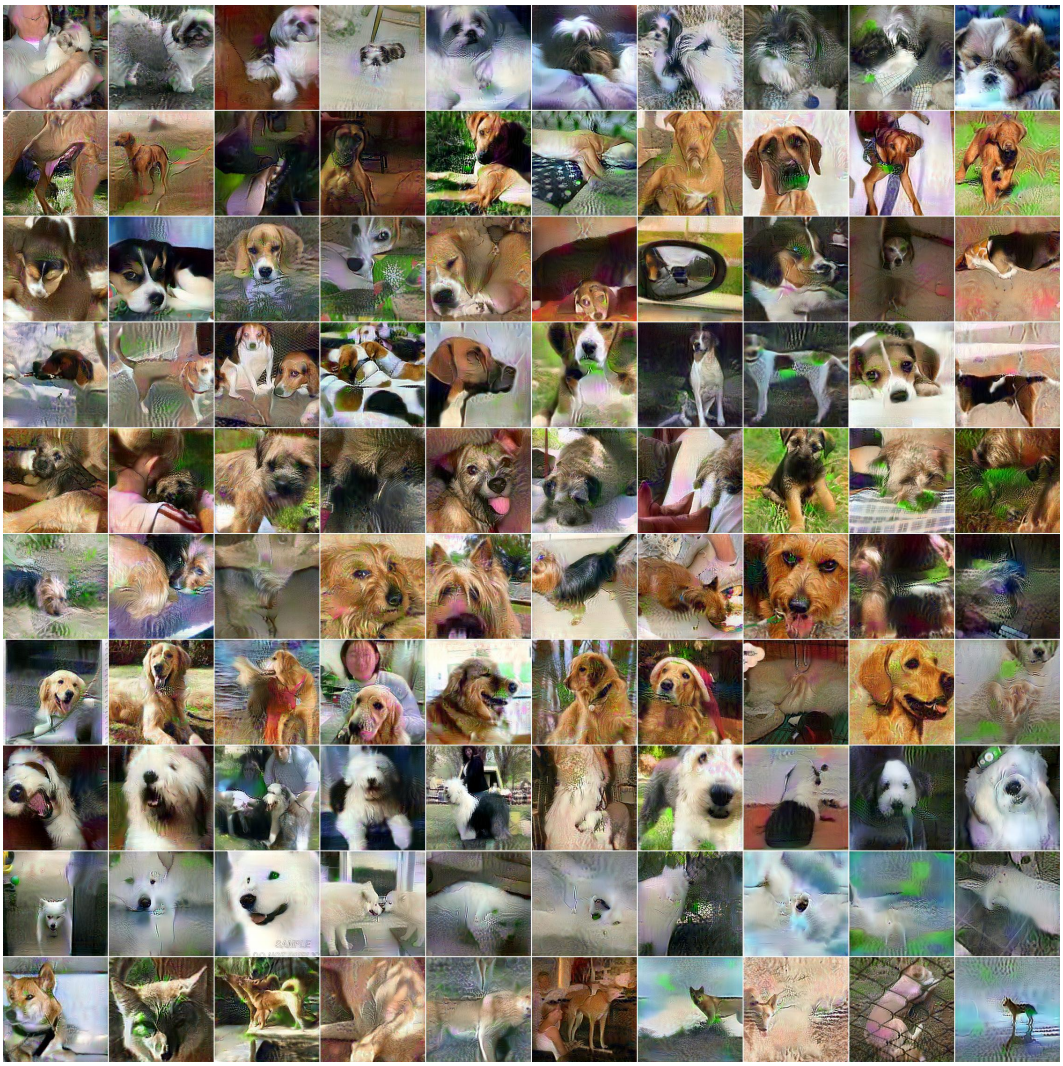


Figure 14: Visualization of the distilled ImageWoof samples generated by PSM with IPC $= 10$ and teacher=ResNet18.