Adaptive Dataset Quantization

Muquan Li, Dongyang Zhang*, Qiang Dong, Xiurui Xie, Ke Qin

Institute of Intelligent Computing, University of Electronic Science and Technology of China, China
muquanli2023@std.uestc.edu.cn, {dyzhang, dongq, xiexiurui, qinke}@uestc.edu.cn

Abstract

Contemporary deep learning, characterized by the training of cumbersome neural networks on massive datasets, confronts substantial computational hurdles. To alleviate heavy data storage burdens on limited hardware resources, numerous dataset compression methods such as dataset distillation (DD) and coreset selection have emerged to obtain a compact but informative dataset through synthesis or selection for efficient training. However, DD involves an expensive optimization procedure and exhibits limited generalization across unseen architectures, while coreset selection is limited by its low data keep ratio and reliance on heuristics, hindering its practicality and feasibility. To address these limitations, we introduce a newly versatile framework for dataset compression, namely Adaptive Dataset Quantization (ADQ). Specifically, we first identify the sub-optimal performance of naive Dataset Quantization (DQ), which relies on uniform sampling and overlooks the varying importance of each generated bin. Subsequently, we propose a novel adaptive sampling strategy through the evaluation of generated bins' representativeness score, diversity score and importance score, where the former two scores are quantified by the texture level and contrastive learning-based techniques, respectively. Extensive experiments demonstrate that our method not only exhibits superior generalization capability across different architectures, but also attains state-of-the-art results.

Code — https://github.com/SLGSP/ADQ

Introduction

Deep learning has witnessed remarkable advancements recently, revolutionizing various tasks in the artificial intelligence community (Ioffe and Szegedy 2015). This progress is primarily attributed to the abundance of datasets with precise labels, which serve as the foundation for training complex models. However, the expanding size of these datasets leads to increased computational costs and resource requirements. This challenge underscores the critical need for efficient dataset compression techniques (Lei and Tao 2024), with focus on reducing the volume of data while ensuring the consistency of training results.

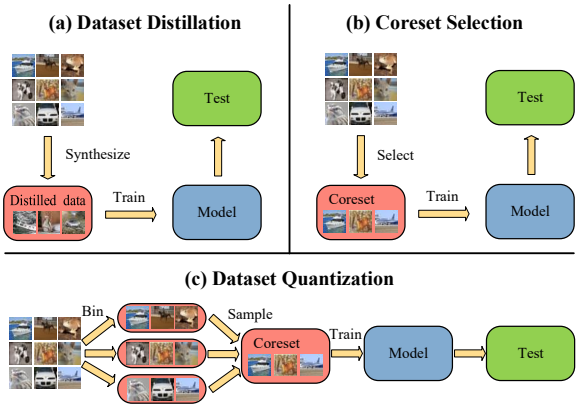
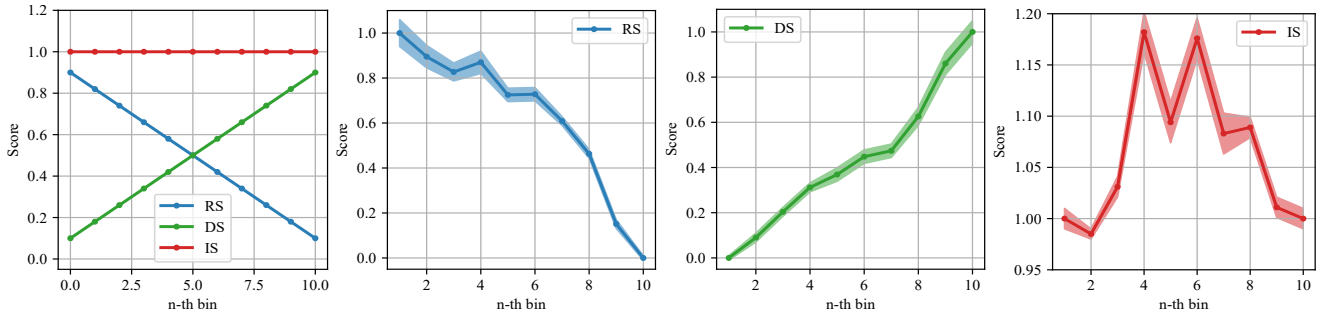


Figure 1: The paradigm of three types of dataset condensation methods. The primary difference between these methods lies in the subset generating process. **(a)** Dataset Distillation synthesizes unreal dataset, **(b)** coreset selection employs one-time selection, while **(c)** dataset quantization utilizes multi-time selection as well as sampling.

In order to improve the computational efficiency, two types of techniques have made great contributions to the dataset compression, namely Dataset Distillation (DD) (Zhao, Mopuri, and Bilen 2021) and coreset selection (Feldman and Zhang 2020). DD has garnered attention for its excellent performance. It aims to generate a compact but informative synthetic dataset, so that models trained with it can attain a similar or even higher level of accuracy. However, the latest optimization-oriented DD methods (Kim et al. 2022; Zhao and Bilen 2023) suffer from high computational costs and poor generalization capability. Specifically, these methods employ a nested loop that alternately optimizes the distilled dataset and pre-trained model parameters(Cazenavette et al. 2022), as well as relying on architecture-driven metrics to align the synthetic samples with the original ones (Zhao, Mopuri, and Bilen 2021; Zhao and Bilen 2023). Consequently, these limitations make it difficult to deploy DD algorithms in real-world scenario and generalize them to other model architectures. Unlike the synthesis of samples for training in DD, coreset selection aims to identify a most important subset from the training set, which has been shown to possess great cross-

*Corresponding Author.



(a) Ideal Condition (b) Representativeness Score (c) Diversity Score (d) Importance Score

Figure 2: The evaluation of normalized representativeness score, diversity score and importance score on CIFAR-10 (Krizhevsky, Hinton et al. 2009). **(a)** Ideal Condition allows for the best performance of DQ. **(b) (c) (d)** are representativeness score (RS), diversity score (DS) and importance score (IS) of generated bins on CIFAR-10, respectively.

architecture generalization capabilities. However, as a traditional dataset compression method that employs a one-time selection strategy, its typically low data keep ratio often fails to preserve the high diversity of the whole dataset, resulting in inferior performance (Zhou et al. 2023). Furthermore, due to its reliance on heuristics, coreset selection cannot guarantee an optimal solution for various downstream tasks (Zhao, Mopuri, and Bilen 2021).

To overcome the limitations of DD and coreset selection, Dataset Quantization (DQ) (Zhou et al. 2023) is a newly proposed pipeline which first partitions the original training dataset by recursively extracting non-overlapping samples into bins based on maximizing submodular gains, and then uniformly sampled from each bin. Fig.1 illustrates the main difference between DQ, DD and coreset selection. Since DQ avoids the dataset synthesis and one-time selection, it can be used for training any model architectures with high data diversity and low computational cost. The sampling strategy in DQ is based on a mathematically derived theory: the bins generated in early steps have a better representativeness of the entire dataset, while the latter bins demonstrate greater diversity. However, the naive DQ does not thoroughly analyze the uneven variations of bins' representiveness and diversity, and overlooks the varying importance of each bin, which in turn impairs the performance.

In this paper, we take a further step based on DQ, through quantitatively evaluating the importance of generated bins and introduce a novel Adaptive Dataset Quantization (ADQ). Specifically, we begin by assessing each bin through three metrics: the Representativeness Score (RS), the Diversity Score (DS), and the Importance Score (IS), which is a composite of RS and DS, corresponding to the theory in DQ. By integrating this theory into sampling strategy, we observe that DQ performs optimally merely under completely ideal condition, where the importance of each bin is equal and the trends of RS and DS resemble the blue and green curves in Fig.2(a). However, under real condition, the paucity of quantitative metrics for RS and DS precludes the appropriate estimation of IS for each bin. Therefore, to provide the evidence for precise sampling in DQ, we define three scores as following:

Representiveness Score. Drawing inspiration from trajectories matching (Cazenavette et al. 2022; Du et al. 2023), we propose a texture level (TL) method to calculate the representativeness score (RS) for real image sets.

Diversity Score. As a precise method for evaluating diversity, contrastive learning-based techniques (Fang et al. 2021) have been proven efficient and cost-effective, for which we introduce to calculate the diversity score (DS) of each bin.

Importance Score. It is intuitive to utilize normalization to combine the representativeness score and diversity score, yielding the importance score (IS) for each bin.

Different from expected ideal condition in DQ, as shown in Fig.2(b)(c)(d), the real condition of these three metrics varies unevenly. It is obvious that uniform sampling strategy in DQ neglects this uneven importance variation of generated bins. Therefore, we adaptively sample from all bins based on the IS of each bin and the amount of the data it contains. Overall, the main contributions of our work can be summarized in the following three aspects:

- We elucidate the sampling limitations of the naive DQ and mathematically establish appropriate metrics for evaluating the representiveness, diversity and importance of the generated bins.
- We propose Adaptive Dataset Quantization (ADQ), which samples data from each generated bin according to its importance score and the number of images, achieving efficient and lossless dataset compression.
- Extensive experiments on CIFAR-10, CIFAR-100 (Krizhevsky, Hinton et al. 2009), ImageNet-1K (Russakovsky et al. 2015) and Tiny-ImageNet (Le and Yang 2015) substantiate a marked enhancement in performance over the baseline DQ by average 3%, establishing the new state-of-the-art results.

Related Works

Dataset Distillation

Dataset Distillation is first proposed in (Wang et al. 2018), where the distilled images are expressed as model weights and optimized by gradient-based hyperparameter tuning. Subsequently, a series of bi-level optimization-oriented

works seek to minimize the surrogate models learned from both synthetic and original datasets, depending on various metrics such as the matching gradients (Zhao, Mopuri, and Bilen 2021; Kim et al. 2022; Zhang et al. 2023), features (Wang et al. 2022), distribution (Zhao and Bilen 2023; Zhao et al. 2023), training trajectories (Cazenavette et al. 2022; Cui et al. 2023; Du et al. 2023), and maximum mean discrepancy (Zhang et al. 2024). However, the synthetic data from these methods often struggle to generalize across different architectures and face significant computational challenges (Zhou et al. 2023). Recently, a notable work (Cazenavette et al. 2023) integrates a plug-and-play module GLaD into existing DD framework to improve generalization, while the high training costs remain a concern. Besides, the unilevel optimization-oriented work (Liu et al. 2022a; Zhou, Nezhadarya, and Ba 2022) effectively reduces calculation costs but may hinder scalability to larger data keep ratio.

Coreset Selection

Coreset selection (Feldman and Zhang 2020; Guo, Zhao, and Bai 2022) focuses on selecting an important subset of the original dataset, showing remarkable potential in facilitating cross-architecture training. To evaluate the subset's importance, multiple metrics have been proposed in previous work: error (Toneva et al. 2019), geometry (Agarwal et al. 2020), memorization (Feldman and Zhang 2020), uncertainty (Coleman et al. 2020), gradient-matching (Killamsetty et al. 2021), submodularity (Iyer et al. 2021), EL2N score (Paul, Ganguli, and Dziugaite 2021), submodular gains (Iyer et al. 2021) and contributing dimension structure (Wan et al. 2024). However, its low data keep ratio leads to impaired diversity of subset (Zhou et al. 2023), and its dependence on heuristics hinders the optimization to downstream task (Zhao, Mopuri, and Bilen 2021). Additionally, as an extension of coreset selection, Dataset Quantization (DQ) (Zhou et al. 2023) improves upon the traditional one-time sampling strategy by recursively generating non-overlapping bins and performing uniform sampling across all bins. This approach enhances the paradigm of sampling by shifting from a single-selection to a multi-selection strategy, thereby maintaining an appropriate data keep ratio in the subset and making it more suitable for various downstream tasks. Nevertheless, the uniform sampling strategy neglects to quantify the importance of the generated bins, for which we chose it as the baseline to address.

Remark

In the realm of dataset compression, previous studies have introduced a variety of metrics to assess the representativeness and diversity of datasets. However, the majority of these methods tend to focus on either representativeness (Iyer et al. 2021) or diversity (Wan et al. 2024), rather than combining both aspects. Additionally, the evaluation techniques are often either overly simplistic, such as using $L2$-norm and cosine distance (Ceccarello, Pietracaprina, and Pucci 2018) to gauge diversity, or excessively comrplex, like utilizing a pre-trained model to derive insights (Li et al. 2018). Typically, these methods depend on a one-time evaluation of the entire dataset, resulting in limited precision. In contrast, we propose a method that employs texture-level analysis and contrastive learning-based techniques to evaluate these metrics for each generated subset. This approach allows us to achieve high precision with low computational demands.

Proposed Method

As mentioned in the Introduction section, we recognize the promising potential of DQ (Zhou et al. 2023) and choose it as the starting point of our research. In this section, we first define the problem that DQ attempts to address. Furthermore, we analyze the clear drawbacks of naive DQ. Finally, we propose three types metrics to evaluate each bin and adaptive sampling to address these drawbacks.

Problem Definition

GraphCut in Coreset Selection Let $\mathbf{D} = \{(x_k, y_k)\}_{k=1}^{M}$ represents M labeled samples. By default, coreset selection involves selecting K samples from $\mathbf{D}$ to form a coreset. The coreset is initialized as $\mathbf{S}_1^1 \leftarrow \emptyset$ and updated as $\mathbf{S}_1^k \leftarrow \mathbf{S}_1^{k-1} \cup x_k$. Note that $\forall p \in \mathbf{D}$, $f(p) \in \mathbb{R}^{m\times 1}$, $\mathbf{S}_n^k$ represents the first k samples of the n-th bin and x_k is the k-th selected sample, $\mathbf{S}_1^{k-1}$ denotes the set of selected samples, $\mathbf{D}\backslash\mathbf{S}_1^{k-1}$ is the remaining set and $f(\cdot)$ is the feature extractor. In GraphCut (GC) (Iyer et al. 2021), samples are selected by maximizing submodular gains $P(x_k)$ in the feature space, defined as follows,

$$P\left(x_k\right) = \sum_{p\in\mathbf{S}_1^{k-1}} \underbrace{\|f(p) - f\left(x_k\right)\|_2^2}_{C_1(x_k)} - \sum_{p\in\mathrm{D}\backslash\mathbf{S}_1^{k-1}} \underbrace{\|f(p) - f\left(x_k\right)\|_2^2}_{C_2(x_k)}. \tag{1}$$

Dataset Quantization Almost all coreset selection methods use a heuristic metric to select samples similar to GC, making it difficult to avoid selecting samples with similar performances according to the metric. To address this selection bias, Dataset Quantization (Zhou et al. 2023) propose a new framework consisting of three steps: bin generation, bin sampling, and pixel quantization. In detail, DQ first partitions the dataset into several non-overlapping bins. Given a dataset $\mathbf{D}$, small informative bins are recursively sampled from $\mathbf{D}$ with a predefined bin size K. Each bin is selected by maximizing the submodular gain described in Eqn.1, resulting in a set of small bins $[\mathbf{S_1}, \ldots, \mathbf{S_n}, \ldots, \mathbf{S_m}]$. The selection of the k-th sample in the n-th bin is formulated as follows,

$$x_k \leftarrow \arg\max \left(\sum_{p\in\mathbf{S_n^{k-1}}} C_1(x_k) - \sum_{p\in\mathbf{D}\backslash\mathbf{S_1}\cup\cdots\cup\mathbf{S_n^{k-1}}} C_2(x_k) \right), \tag{2}$$

where $C_1\left(x_k\right)$ and $C_2\left(x_k\right)$ have been defined in Eqn. 1, $\mathbf{D}\setminus(\mathbf{S_1}\cup\cdots\cup\mathbf{S_n^{k-1}})$ represents the remaining data in the dataset after selecting $(k-1)$ samples in n-th bin.

Following this, a uniform sampler $g(\cdot,\cdot)$ is used to sample a specific portion from each bin to form the final coreset set. Additionally, inspired by reconstructing images using only some of their patches in the Masked Auto-Encoder (MAE) (He et al. 2022), DQ discards less important patches to reduce the number of pixels used for describing each image. The detailed patch dropping and reconstruction strategy is described in the Appendix.

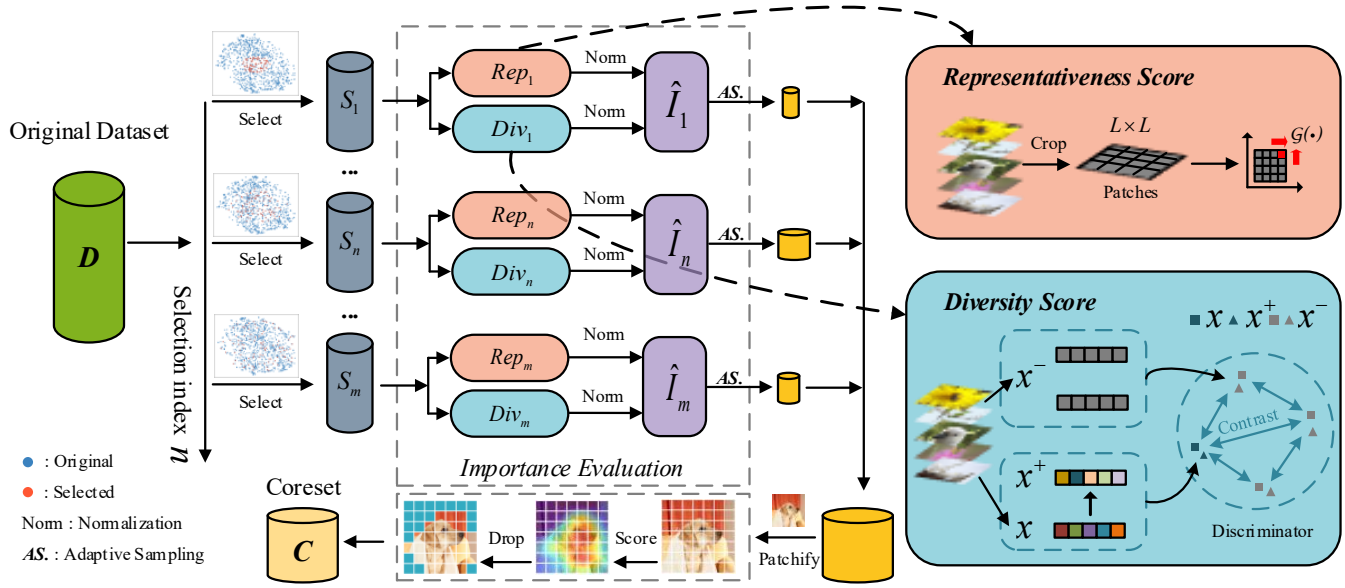


Figure 3: The overall framework of the proposed Adaptive Dataset Quantization (ADQ). Following Dataset Quantization (DQ), we first divide the original dataset $\mathbf{D}$ into m non-overlapping bins $[\mathbf{S}_1, ...\mathbf{S}_n, ...\mathbf{S}_m]$. Next, an importance evaluation is conducted to calculate representativeness score, diversity score and importance score for $\mathbf{S_n}$. We then employ an adaptive sampling based on the importance score and the number of samples in $\mathbf{S_n}$ to obtain a initial compressed set. Eventually, a patch dropping and reconstruction process via MAE (He et al. 2022) is used to drop uninformative patches, as detailed in the Appendix.

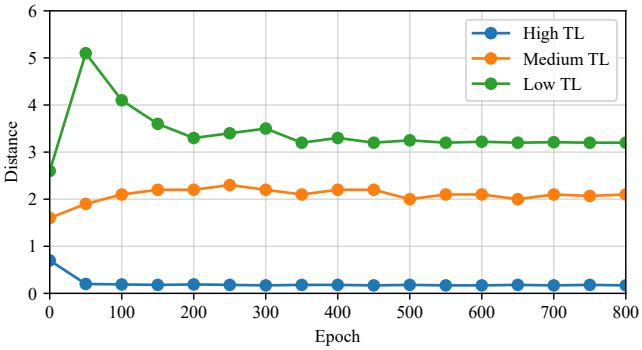


Figure 4: The illustration of three types of texture level (TL) curves: High TL, Medium TL and Low TL. These curves represent the distances between the expert and the students in our improved trajectories matching.

Problem Analysis

Although DQ achieves high coverage of the overall data across different model architectures, it encounters a significant challenge. According to the derivation of average feature (Zhou et al. 2023), the bin generated in the earlier steps is primarily influenced by the distances within the remaining data, while the bin in the later steps is more affected by the diversity of data in the current bin. To balance representativeness and diversity, DQ employs simple uniform sampling to randomly select an equal proportion of data from each bin. However, this uniform sampling strategy performs optimally only under an ideal condition. Specifically, given that the representativeness and diversity of each bin are unknown, their importance for inclusion in the original dataset remains uncertain. If the influence of representativeness and diversity on the results does not exhibit a uniform linear variation, as shown in Fig.2(a), then uniform sampling may only achieve a spurious balance and fail to produce the best possible outcomes.

Importance Evaluation

Obviously, evaluating the varying importance of the sequentially generated bins is crucial for rectifying this spurious balance. To effectively illustrate the variation in importance, we quantify three metrics for each bin, as follow:

Representativeness Score Inspired by the trajectories matching (Du et al. 2023), quantifying the representativeness of each bin can be approached by calculating the distance between each bin and the original dataset along different training trajectories. We theoretically assume the existence of an expert parameter representing the optimal training trajectory, which corresponds to the training trajectory of the original dataset. Other training trajectories are considered student parameters. The distance between the expert and student is then calculated in the parameter space to reflect the representativeness of different bins for the entire dataset. However, traditional trajectory matching methods (Cazenavette et al. 2022; Du et al. 2023) are typically optimized through backpropagation on non-real images during dataset distillation, which contrasts with DQ that operates on real images.

Addressing this limitation, we propose a straightforward yet effective technique, termed the texture level (TL)

method, as an alternative to utilizing training trajectories for trajectory matching in real images. Specifically, we first crop the images in each bin into patches P of size $L \times L$. Following this, we introduce a general gradient operator $\mathcal{G}(\cdot)$ to calculate the texture level $T(\cdot)$ of each bin:

$$T(P) = \frac{1}{L^2} \sum_{i,j \in [1,\dots,L]} \mathcal{G}(P_{i,j}), \tag{3}$$

where the subscript i, j denotes the pixel coordinates.

To demonstrate the matching effect of texture level, we then crop the entire original dataset into patches and calculate the texture level of each patch. These patches are divided into three equal batches, each representing a third of the dataset: the top third are classified as High Texture Level, the middle third as Medium Texture Level, and the bottom third as Low Texture Level. Next, we train the selected model on these three batches (as the students) and on the original dataset (as the expert), while calculating a type of trajectory parameter distance in each batch with original dataset. Details about trajectory parameter distance are provided in Appendix. Intuitively, we obtain three distance curves varying with training epochs. Fig.4 is obtained by training ResNet-18 on the CIFAR-10 dataset. It is observed that the distance between the student model and the expert model decreases as texture level increasing. For models trained on high-level texture patches, this distance approaches zero, indicating that images with more complex textures guide the model to progress along a trajectory more similar to that of the original dataset. Therefore, we transform the calculation of the RS $Rep(\cdot)$ for each bin into the computation of its texture level $T(\cdot)$, through $Rep(\cdot) = T(\cdot)$.

Diversity Score We introduce a contrastive learning-based method for measuring diversity (Fang et al. 2021), modeling data diversity as an instance discrimination problem. First, we introduce a discriminator $d(\cdot)$, which is a simple multi-layer perception that takes the representation from the penultimate layer and the global pooling of intermediate features as input. In each bin $\mathbf{S}_n$, a positive view x^+ is constructed for each image using random augmentation, such as rotations, flips, and color adjustments, to enhance variability, while other images in $\mathbf{S}_n$ are considered negative views x^-. The discriminator learns to distinguish different samples by pulling positive samples closer and pushing negative samples farther apart, thereby calculating data diversity through contrastive learning. We use simple cosine similarity $cos(\cdot)$ to describe the relationship between data pairs x_1 and x_2:

$$cos(x_1, x_2, d) = \frac{\langle d(x_1), d(x_2) \rangle}{\|d(x_1)\| \cdot \|d(x_2)\|}, \tag{4}$$

Let τ be the temperature parameter of the discriminator. The diversity $Div(\cdot)$ of data for $\mathbf{S}_n$ can then be represented as follows:

$$\begin{aligned} Div(\mathbf{S}_n) &= -\mathbb{E}_{x_i \in \mathbf{S}_n} \left[\mathbb{E}_{x_j \in \mathbf{S}_n} \left[\frac{\exp(cos(x_i, x_j^-, d)/\tau)}{\exp(cos(x_i, x_i^+, d)/\tau)} \right] \right] \\ &= -\frac{1}{N(x^-)} \left[\mathbb{E}_{x_i \in \mathbf{S}_n} \left[\frac{\sum_j \exp(cos(x_i, x_j, d)/\tau)}{\exp(cos(x_i, x_i^+, d)/\tau)} \right] \right], \end{aligned} \tag{5}$$

where $N(x^-)$ refers to the amount of negative samples for each x_i in $\mathbf{S}_n$.

Importance Score After calculating the RS and DS, we normalize (Ioffe and Szegedy 2015) both scores separately to facilitate the evaluation of the varying importance of generated bins on the same scale:

$$\hat{Rep}_n = \text{Norm}(Rep_n, \mathbb{S}_{Rep}), \tag{6}$$

$$\hat{Div}_n = \text{Norm}(Div_n, \mathbb{S}_{Div}), \tag{7}$$

where Rep_n and Div_n represent the RS and DS of the n-th bin, $\mathbb{S}_{Rep}$ and $\mathbb{S}_{Div}$ denote the sets of all RS and DS. We then defined the IS $\hat{I}_n$ for n-th bin as the sum of the normalised RS and DS:

$$\hat{I}_n = \hat{Rep}_n + \hat{Div}_n, \tag{8}$$

The variations in RS, DS, and IS of bins during the bin generation process are illustrated in Fig.2(b)(c)(d). As observed, the overall importance of bins initially increases and then decreases throughout the generation process. This pattern corresponds with our analysis of the spurious balance discussed in the Problem Analysis section. To capitalize on the dynamic importance of each bin, we introduce an adaptive sampling method.

Adaptive Sampling

We calculate the proportion r_n of images to be selected from each bin based on its normalized importance value $\hat{I}_n$ and the number of images $N(n)$ in the n-bin:

$$r_n = \alpha \hat{I}_n + (1 - \alpha) \frac{N(n)}{\sum_{n=1}^{m} N(n)}, \tag{9}$$

where $\alpha \in [0, 1]$ denotes a weighting coefficient to balance the importance and the number of images in each bin, m represents the total number of generated bins. The effect of value α will be discussed in Ablation Study section. Eventually, the final number of images q_n selected from each bin is determined:

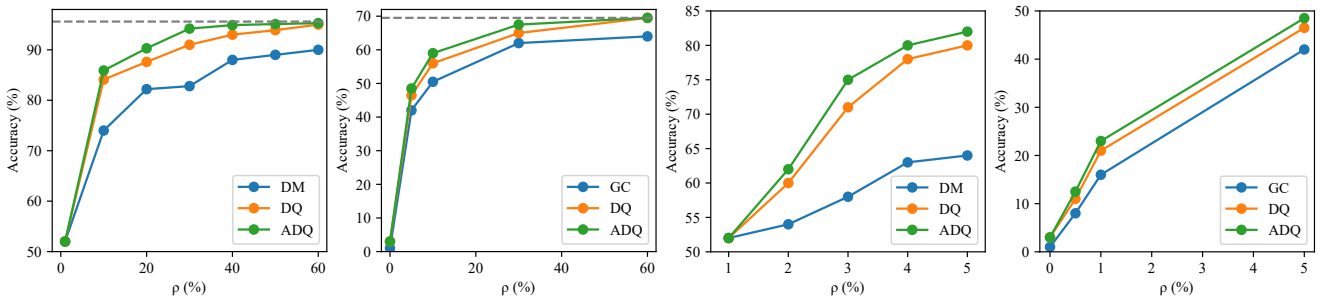
$$q_n = \lfloor r_n \times N(n) \rfloor, \tag{10}$$

where $\lfloor \cdot \rfloor$ denotes the floor function, ensuring that the total number of selected images does not exceed the required number. The adaptive process after bin generation is shown in Alg.1. Following this, a process of patch dropping and reconstruction is used to remove invalid information (He et al. 2022), as detailed in the Appendix. The overall framework of our ADQ is illustrated in Fig.3.

Experiments

Experimental Setup

Datasets Following the evaluation protocol of previous DQ (Zhou et al. 2023), we utilize image classification as a proxy task for evaluation and mainly assess our method on CIFAR-10 (Krizhevsky, Hinton et al. 2009) and ImageNet-1K (Russakovsky et al. 2015). CIFAR-10 contains 50,000 samples for training and 10,000 samples for validation, with a resolution of 32×32. ImageNet-1K comprises 128,1126 samples from 1000 categories for training, with each category containing 50 images for validation.



(a) High Ratio (b) High Ratio (c) Low Ratio (d) Low Ratio

Figure 5: The performance of DM (Zhao and Bilen 2023), DQ (Zhou et al. 2023) and ADQ on **(a)** high data keep ratio and **(c)** low data keep ratio on CIFAR-10; and GC (Iyer et al. 2021), DQ and ADQ on **(b)** high data keep ratio and **(d)** low data keep ratio on ImageNet-1K. The dashed lines in grey in **(a)** and **(b)** indicate the results when the data keep ratio is 100%.

Algorithm 1: Adaptive Dataset Quantization

Input: m dataset bins $\mathbf{S}_1, ...\mathbf{S}_n, ...\mathbf{S}_m$.

Required: Patch size $L \times L$, temperature parameter τ of the discriminator, weighting coefficient α.

for $n = 1, ..., m$ **do**

- P with $L \times L \leftarrow \mathbf{S}_n$
- Calculate Rep_n using Eqn.3
- $x^+, x^- \leftarrow x$ in $\mathbf{S}_n$
- Calculate Div_n using Eqn.4
- $\hat{Rep}_n \leftarrow \text{Norm}(Rep_n)$; $\hat{Div}_n \leftarrow \text{Norm}(Div_n)$
- $\hat{I}_n \leftarrow \hat{Rep}_n + \hat{Div}_n$
- Calculate r_n using Eqn.9
- Calculate q_n using Eqn.10
- Select randomly q_n samples from n-th bin

Output: Initial compressed dataset.

Implementation details Unless specified, we mainly use the ResNet-18 (He et al. 2016) and Vision Transformer (ViT-base) (Dosovitskiy et al. 2021) models as the feature extractor for CIFAR-10 and ImageNet-1K, respectively. To assess the generalization of the compressed dataset, the training processes are implemented on several representative transformer and CNN architectures, including ResNet-18, ResNet-50 (He et al. 2016), ViT, Swin transformer (Liu et al. 2021), ConvNeXt (Liu et al. 2022b) and MobilenetV2 (Sandler et al. 2018). During bin generation, the experimental procedure is consistent with those in DQ (Zhou et al. 2023). For comparison, we conduct training for 200 epochs on the CIFAR-10 with batch size 128, and we employ a cosine-annealed learning rate that initializes at 0.1. For ImageNet-1K, the training is in Distributed Data Parallel manner with the default scripts for different architectures. We conduct 5 experiments to average the results. For more details about the reproduction of the paper, please refer to the Appendix.

Comparisons with Previous Methods

Tab.1 and Fig.5(a) present a comparison of our method with previous DM (Zhao and Bilen 2023) and DQ (Zhou et al. 2023) on CIFAR-10 dataset. DM is the pioneering method that approaches data condensation via distribution matching. DQ is the first method to divide the full distribution into non-overlapping bins and then uniformly sampling from each bin, working as our baseline. In line with DQ, we use three data keep ratios (10%, 20%, and 30%) to evaluate the performance variations, in addition to the 100% ratio for a comprehensive comparison. The results reveal that datasets generated from DQ and ADQ retain higher performance levels when tested with new architectures during training. Notably, our ADQ consistently outperforms DQ across all five architectures, with average improvements of 2.6%, 2.8%, and 3.3% at these ratios, respectively. The performance gains with higher data keep ratios are attributed to the increased number of effective samples available for calculating RS and DS, which in turn enhances the accuracy of the IS. For ImageNet-1K, we substitute DM with GraphCut (GC) (Iyer et al. 2021), and observe similar performance improvements with ADQ, as illustrated in Fig.5(b).

Following DQ, we extend our performance comparisons to low data keep ratios to further highlight the metrics of ADQ, as depicted in Fig.5(c)(d). For lossless compression, our ADQ also achieves lossless results with only 60% of the data, matching the performance of the current state-of-the-art dataset compression methods (Zhou et al. 2023). Turning the attention to the practical aspects of dataset generation, Tab.2 provides a comparison of our ADQ with DM and DQ in terms of the number of runs, error bars, and GPU hours required. Our ADQ exhibits a reduction in average error bars across all experimental conditions. Notably, the computational modules we introduce for importance evaluation contribute negligible additional processing time. As a result, the time ADQ requires for dataset generation is on par with that of DQ and is a mere 1.1% of the time needed by DM, underscoring ADQ's efficiency.

Ablation Study

Module Cut-off The ablation study begins by evaluating the contributions of the proposed three metrics of the bin: RS, DS and IS. As shown in Tab.3, DQ serves as our baseline, and its performance on CIFAR-10 is intuitively enhanced by incorporating RS, DS and IS, with averages improvements of 1.41%, 1.53% and 2.57%, respectively. Note

	DM				DQ				ADQ			
ρ (%)	10	20	30	100	10	20	30	100	10	20	30	100
ResNet-18	74.0	82.2	82.8	95.6	84.1	87.6	91.0	95.6	86.2 (+2.1)	90.4 (+2.8)	94.2 (+3.2)	95.6
ResNet-50	35.0	36.2	43.9	95.5	82.7	88.1	90.8	95.5	84.7 (+2.0)	90.7 (+2.6)	93.7 (+2.9)	95.5
ViT	21.6	25.5	23.1	80.2	58.4	66.8	72.0	80.2	61.1 (+2.7)	69.8 (+3.1)	74.7 (+2.7)	80.2
Swin	25.1	30.1	27.3	90.3	69.2	79.1	84.4	90.3	73.2 (+4.0)	82.5 (+3.4)	88.5 (+4.1)	90.3
ConvNeXt	41.8	48.3	47.9	73.0	52.8	61.8	64.2	73.0	55.0 (+2.2)	64.0 (+2.2)	68.1 (+3.9)	73.0
Average	39.5	44.5	45	86.9	69.4	76.7	80.5	86.9	**72.0 (+2.6)**	**79.5 (+2.8)**	**83.8 (+3.3)**	86.9

Table 1: Comparisons of DM (Zhao and Bilen 2023), DQ (Zhou et al. 2023) and our ADQ on CIFAR-10 with different data keep ratios ρ. The training processes are implemented across five various architectures, with ResNet-18 used as the feature extractor to obtain distilled data. Each reported result is the average of 5 experiments.

Method	Number of runs	Error bars	GPU hours
DM	5	± 0.5	$91h$
DQ	3	± 0.4	$1h$
ADQ	5	± 0.2	$1h$

Table 2: Comparisons of number of runs, error bars and GPU hours for compressing dataset of DM, DQ and ADQ.

Dataset	CIFAR-10			
ρ (%)	10	20	30	100
DQ	84.1	87.6	91.0	95.6
+ RS	85.1	88.8	93.0	95.6
+ DS	85.3	88.9	93.1	95.6
+ IS (RS+DS)	86.2	90.4	94.2	95.6

Table 3: Ablation study on RS, DS and IS, training on CIFAR-10. DQ presents our baseline. The increasing accuracy of results with incorporating three modules demonstrates the effectiveness of our ADQ.

that the improvement of DS is slightly higher than that of RS across all data keep ratios, which suggests that diversity plays a more critical role than representativeness in impacting the final performance of the subset. Nevertheless, given the fluctuating conditions across different datasets and the potential rise in computational complexity due to tuning the weight ratio between representativeness and diversity (yielding only slight improvements), we maintain equal weighting for both factors when computing the importance score.

Hyper-parameter analysis There are three hyper-parameters for ADQ: the numbers of bins m, the drop ratio θ and the weighting coefficient α, where the first two parameters have been proven to give the optimal trade-off with $m = 10$ and $\theta = 25\%$ in DQ. Fig.6 illustrates how the performance of our ADQ varies with different choices of α. We conduct the data-keep-ratio-dependent experiments on CIFAR-10 cross two architectures, ResNet-18 and ResNet-50. As observed, accuracy initially increases and then decreases as α ranges from 0 to 1, reaching its peak between 0.6 and 0.7. Interestingly, the peaks of accuracy on both two datasets are shifting back (closer to 0.7).

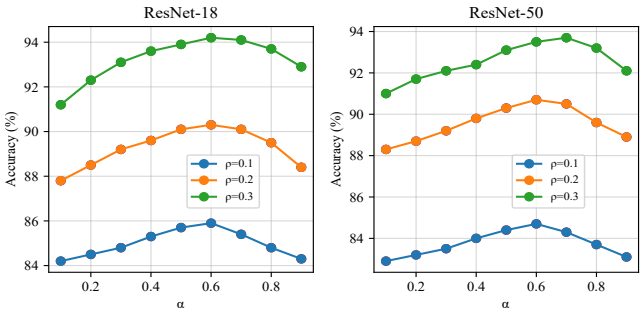


Figure 6: Ablation study on the weighting coefficient α. ResNet-18 and ResNet-50 are utilized as training models, and the experiments are implemented on three different values of data keep ratio. The average accuracy is reported.

Given α presents the weighting of the importance score in normalized importance score (Eqn.9), we ascribe this trend to the increased number of evaluation bases, where higher data keep ratio provides more data for assessing the importance score. As α approaches 0, ADQ reverts to DQ, resulting in performance that mirrors that of DQ. During the actual experiment, we adjust corresponding values of α according to different architectures.

Conclusion

In this paper, we introduce an Adaptive Dataset Quantization (ADQ) approach designed to address the suboptimal performance of the naive DQ method, which overlooks the differing significance of the produced bins. Specifically, we delineate three metrics for each bin: the RS, the DS, and the IS. We then employ a texture-level method and a contrastive learning-based method to compute the RS and DS, respectively. Ultimately, the IS is obtained by integrating the RS and DS, which facilitates ADQ based on the bin's importance. Extensive experimental results confirm the efficacy of our ADQ, showing a comprehensive enhancement over the naive DQ. For future research, we intend to investigate the application of ADQ in various downstream tasks, such as object detection, image restoration.

Acknowledgments

This work is partially supported by grants from the China Postdoctoral Science Foundation (No.2024M760357), the Postdoctoral Fellowship Program of CPSF (No.GZB20240115), Sichuan Central-Guided Local Science and Technology Development (No.2023ZYD0165), the National Natural Science Foundation of China (NO.62176046) and Noncommunicable Chronic Diseases-National Science and Technology Major Project (No.2023ZD0501806).

References

Agarwal, S.; Arora, H.; Anand, S.; and Arora, C. 2020. Contextual Diversity for Active Learning. In *Computer Vision - ECCV 2020 - 16th European Conference ECCV*, volume 12361, 137–153. Springer.

Cazenavette, G.; Wang, T.; Torralba, A.; Efros, A. A.; and Zhu, J. 2022. Dataset Distillation by Matching Training Trajectories. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 10708–10717. IEEE.

Cazenavette, G.; Wang, T.; Torralba, A.; Efros, A. A.; and Zhu, J. 2023. Generalizing Dataset Distillation via Deep Generative Prior. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 3739–3748. IEEE.

Ceccarello, M.; Pietracaprina, A.; and Pucci, G. 2018. Fast Coreset-based Diversity Maximization under Matroid Constraints. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM*, 81–89. ACM.

Coleman, C.; Yeh, C.; Mussmann, S.; Mirzasoleiman, B.; Bailis, P.; Liang, P.; Leskovec, J.; and Zaharia, M. 2020. Selection via Proxy: Efficient Data Selection for Deep Learning. In *8th International Conference on Learning Representations, ICLR*. OpenReview.net.

Cui, J.; Wang, R.; Si, S.; and Hsieh, C. 2023. Scaling Up Dataset Distillation to ImageNet-1K with Constant Memory. In *International Conference on Machine Learning, ICML*, volume 202, 6565–6590. PMLR.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *9th International Conference on Learning Representations, ICLR*. OpenReview.net.

Du, J.; Jiang, Y.; Tan, V. Y. F.; Zhou, J. T.; and Li, H. 2023. Minimizing the Accumulated Trajectory Error to Improve Dataset Distillation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 3749–3758. IEEE.

Fang, G.; Song, J.; Wang, X.; Shen, C.; Wang, X.; and Song, M. 2021. Contrastive Model Invertion for Data-Free Knolwedge Distillation. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI*, 2374–2380.

Feldman, V.; and Zhang, C. 2020. What Neural Networks Memorize and Why: Discovering the Long Tail via Influence Estimation. In *A Annual Conference on Neural Information Processing Systems, NeurIPS*.

Guo, C.; Zhao, B.; and Bai, Y. 2022. DeepCore: A Comprehensive Library for Coreset Selection in Deep Learning. In *Database and Expert Systems Applications - 33rd International Conference, DEXA*, volume 13426, 181–195. Springer.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. B. 2022. Masked Autoencoders Are Scalable Vision Learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 15979–15988. IEEE.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 770–778. IEEE Computer Society.

Ioffe, S.; and Szegedy, C. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Bach, F. R.; and Blei, D. M., eds., *Proceedings of the 32nd International Conference on Machine Learning, ICML*, volume 37, 448–456.

Iyer, R. K.; Khargoankar, N.; Bilmes, J. A.; and Asanani, H. 2021. Submodular combinatorial information measures with applications in machine learning. In *Algorithmic Learning Theory*, volume 132, 722–754. PMLR.

Killamsetty, K.; Sivasubramanian, D.; Ramakrishnan, G.; De, A.; and Iyer, R. K. 2021. GRAD-MATCH: Gradient Matching based Data Subset Selection for Efficient Deep Model Training. In *Proceedings of the 38th International Conference on Machine Learning, ICML*, volume 139, 5464–5474. PMLR.

Kim, J.; Kim, J.; Oh, S. J.; Yun, S.; Song, H.; Jeong, J.; Ha, J.; and Song, H. O. 2022. Dataset Condensation via Efficient Synthetic-Data Parameterization. In *International Conference on Machine Learning, ICML*, volume 162, 11102–11118. PMLR.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images. *Technical report Citeseer*.

Le, Y.; and Yang, X. 2015. Tiny imagenet visual recognition challenge. *Technical Report*, 7(7): 3.

Lei, S.; and Tao, D. 2024. A Comprehensive Survey of Dataset Distillation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(1): 17–32.

Li, J.; Cheng, K.; Wang, S.; Morstatter, F.; Trevino, R. P.; Tang, J.; and Liu, H. 2018. Feature Selection: A Data Perspective. *ACM Comput. Surv.*, 50(6): 94:1–94:45.

Liu, Z.; Hu, H.; Lin, Y.; Yao, Z.; Xie, Z.; Wei, Y.; Ning, J.; Cao, Y.; Zhang, Z.; Dong, L.; Wei, F.; and Guo, B. 2022a. Swin Transformer V2: Scaling Up Capacity and Resolution. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 11999–12009. IEEE.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV*, 9992–10002. IEEE.

Liu, Z.; Mao, H.; Wu, C.; Feichtenhofer, C.; Darrell, T.; and Xie, S. 2022b. A ConvNet for the 2020s. In *IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, CVPR, 11966–11976. IEEE.

Paul, M.; Ganguli, S.; and Dziugaite, G. K. 2021. Deep Learning on a Data Diet: Finding Important Examples Early in Training. In *Annual Conference on Neural Information Processing Systems, NeurIPS*, 20596–20607.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M. S.; Berg, A. C.; and Fei-Fei, L. 2015. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.*, 115(3): 211–252.

Sandler, M.; Howard, A. G.; Zhu, M.; Zhmoginov, A.; and Chen, L. 2018. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 4510–4520.

Toneva, M.; Sordoni, A.; des Combes, R. T.; Trischler, A.; Bengio, Y.; and Gordon, G. J. 2019. An Empirical Study of Example Forgetting during Deep Neural Network Learning. In *7th International Conference on Learning Representations, ICLR*. OpenReview.net.

Wan, Z.; Wang, Z.; Wang, Y.; Wang, Z.; Zhu, H.; and Satoh, S. 2024. Contributing Dimension Structure of Deep Feature for Coreset Selection. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI*, 9080–9088. AAAI Press.

Wang, K.; Zhao, B.; Peng, X.; Zhu, Z.; Yang, S.; Wang, S.; Huang, G.; Bilen, H.; Wang, X.; and You, Y. 2022. CAFE: Learning to Condense Dataset by Aligning Features. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022*, 12186–12195. IEEE.

Wang, T.; Zhu, J.; Torralba, A.; and Efros, A. A. 2018. Dataset Distillation. *CoRR*, abs/1811.10959.

Zhang, H.; Li, S.; Wang, P.; Zeng, D.; and Ge, S. 2024. M3D: Dataset Condensation by Minimizing Maximum Mean Discrepancy. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI*, 9314–9322. AAAI Press.

Zhang, L.; Zhang, J.; Lei, B.; Mukherjee, S.; Pan, X.; Zhao, B.; Ding, C.; Li, Y.; and Xu, D. 2023. Accelerating Dataset Distillation via Model Augmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023*, 11950–11959. IEEE.

Zhao, B.; and Bilen, H. 2023. Dataset Condensation with Distribution Matching. In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV*, 6503–6512. IEEE.

Zhao, B.; Mopuri, K. R.; and Bilen, H. 2021. Dataset Condensation with Gradient Matching. In *9th International Conference on Learning Representations, ICLR*. OpenReview.net.

Zhao, G.; Li, G.; Qin, Y.; and Yu, Y. 2023. Improved Distribution Matching for Dataset Condensation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 7856–7865. IEEE.

Zhou, D.; Wang, K.; Gu, J.; Peng, X.; Lian, D.; Zhang, Y.; You, Y.; and Feng, J. 2023. Dataset Quantization. In *IEEE/CVF International Conference on Computer Vision, ICCV*, 17159–17170. IEEE.

Zhou, Y.; Nezhadarya, E.; and Ba, J. 2022. Dataset Distillation using Neural Feature Regression. In *Annual Conference on Neural Information Processing Systems NeurIPS*.