Post Training Quantization for Efficient Dataset Condensation

Linh-Tam Tran, and Sung-Ho Bae*

Kyung Hee University, Republic of Korea
{tamlt,shbae}@khu.ac.kr

Abstract

Dataset Condensation (DC) distills knowledge from large datasets into smaller ones, accelerating training and reducing storage requirements. However, despite notable progress, prior methods have largely overlooked the potential of quantization for further reducing storage costs. In this paper, we take the first step to explore post-training quantization in dataset condensation, demonstrating its effectiveness in reducing storage size while maintaining representation quality without requiring expensive training cost. However, we find that at extremely low bit-widths (e.g., 2-bit), conventional quantization leads to substantial degradation in representation quality, negatively impacting the networks trained on these data. To address this, we propose a novel patch-based post-training quantization approach that ensures localized quantization with minimal loss of information. To reduce the overhead of quantization parameters, especially for small patch sizes, we employ quantization-aware clustering to identify similar patches and subsequently aggregate them for efficient quantization. Furthermore, we introduce a refinement module to align the distribution between original images and their dequantized counterparts, compensating for quantization errors. Our method is a plug-and-play framework that can be applied to synthetic images generated by various DC methods. Extensive experiments across diverse benchmarks including CIFAR-10/100, Tiny ImageNet, and ImageNet subsets demonstrate that our method consistently outperforms prior works under the same storage constraints. Notably, our method doubles the test accuracy of existing methods at extreme compression regimes (e.g., from 26.0% to 54.1% for DM at IPC=1), while operating directly on 2-bit images without additional distillation.

Introduction

Deep Neural Networks (DNNs) have become a primary solution in many computer vision tasks (He et al. 2015; Redmon et al. 2016). However, training these networks effectively requires vast amounts of data and significant computational resources (Simonyan and Zisserman 2015; He et al. 2015; Redmon et al. 2016). To address this, Dataset Condensation (DC) (Wang et al. 2018; Zhao, Mopuri, and Bilen 2021) techniques have been developed to create smaller, condensed datasets that preserve the performance of the original, larger dataset. Due to its effectiveness, DC is an active research topic (Zhao, Mopuri, and Bilen 2021; Cazenavette et al. 2022; Wang et al. 2022; Yuan et al. 2024) with numerous applications, including neural architecture search (Zhao, Mopuri, and Bilen 2021), continual learning (Yang et al. 2023; Gu et al. 2024), super-resolution (Zhang et al. 2024), and privacy preserving (Dong, Zhao, and Lyu 2022; Chai et al. 2024).

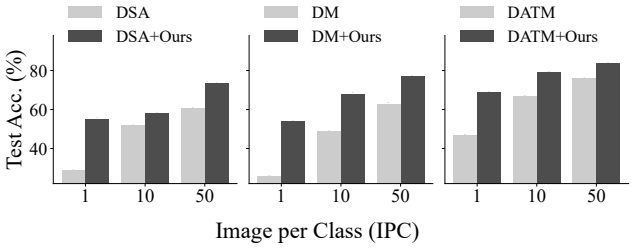


Figure 1: Performance comparison when applying our framework to DSA (Zhao and Bilen 2021), DM (Zhao and Bilen 2022), and DATM (Guo et al. 2024). Notably, our framework **doubles the performance** for DM at a budget of 1 Image Per Class (IPC), demonstrating its effectiveness in extremely low-storage scenarios.

Approaches such as gradient matching (Zhao, Mopuri, and Bilen 2021), distribution matching (Zhao and Bilen 2022), and trajectory matching (Cazenavette et al. 2022) have demonstrated strong performance in producing high-quality condensed datasets. However, these methods primarily focus on improving dataset quality while neglecting storage efficiency, as each synthetic sample still requires full-resolution storage.

To overcome such storage bottlenecks, Parameterization-based Dataset Condensation (PDC) methods encode synthetic data as compact latent representations instead of raw pixels (Yuan et al. 2024). These include frequency-space learning (Shin, Shin, and Moon 2023), sample factorization (Liu et al. 2022), and color space optimization (Yuan et al. 2024). By reconstructing images at inference time, PDC methods achieve higher compression while retaining task-relevant information, often surpassing conventional DC in efficiency and scalability.

*Corresponding Author

However, to the best of our knowledge, no prior work has explored post-training quantization (PTQ) to compress synthetic images for dataset condensation, despite its potential to significantly reduce storage costs with minimal overhead. This motivates our work, which aims to integrate PTQ into DC, reducing the bit-width of stored synthetic images without requiring expensive retraining or decoder networks.

To this end, we propose a novel framework that stores synthetic images in extremely low bit-widths (*e.g.*, 2-bit). A naive application of global quantization—using a single set of quantization parameters for an entire image—suffers from severe degradation of task-relevant information due to spatially varying textures and details being quantized with insufficient precision. This results in a dramatic drop in the network's performance when trained on the dequantized images.

To address this, we propose a *patch-based asymmetric quantization* scheme that applies quantization locally to each image patch, effectively preserving spatial variations. To further reduce the overhead of quantization parameters, we introduce a quantization-aware grouping strategy that clusters similar patches and applies shared quantization within each group. Finally, we present a refinement module that fine-tunes the synthetic images to align their feature distributions with those of the original images, mitigating the effects of quantization noise.

By applying our method to various synthetic images generated from existing DC techniques, we observe up to **2**$\times$ **accuracy improvement at IPC=1** (see Figure 1) under extreme storage constraints, while operating entirely with 2-bit images.

Our contributions are summarized as follows:

- We introduce a patch-based quantization method that significantly improves task performance at extremely low bit-widths by preserving spatial and structural details.
- We propose a clustering-based grouping strategy to share quantization parameters among similar patches, reducing storage overhead.
- We present a quantization-aware refinement module that aligns the distributions of dequantized and original synthetic images to improve downstream model performance.
- We validate our method through extensive experiments on diverse datasets, consistently achieving state-of-the-art performance under constrained storage budgets.

Related Works

Dataset Condensation and Redundancy

Dataset Condensation (DC) aims to distill a large dataset into a compact synthetic set such that a model trained on it performs comparably to one trained on the full dataset (Wang et al. 2018; Zhao, Mopuri, and Bilen 2021). Early approaches address this by surrogate optimization, such as matching the gradients (Zhao, Mopuri, and Bilen 2021; Kim et al. 2022; Zhao and Bilen 2021), network parameters (Cazenavette et al. 2022; Cui et al. 2023; Du et al. 2023; Guo et al. 2024), or intermediate feature distributions (Zhao and Bilen 2022; Zhao et al. 2023; Wang et al. 2022; Sajedi et al. 2023) between real and synthetic data.

Method	Redundancy	Strategy	Bit	Cost
IDC	Spatial	Spatial Downsample	32	Med.
AutoPalette	Color	Palette Reduction	32	High
DDiF	Feature	Neural Field	32	High
Ours	Bit-level	Patch Quantization	**2**	**Low**

Table 1: Comparison of redundancy reduction strategies in PDC.

While these methods successfully preserve task performance, they are inefficient in storage since the synthesized images are stored at full resolution and precision. To overcome this, PDC explores compact representations by reducing various forms of redundancy:

- **IDC** reduces *spatial redundancy* by storing data in low-dimensional latent spaces and reconstructing them via upsampling (Kim et al. 2022).
- **AutoPalette** targets *color redundancy* using a learned palette encoder (Yuan et al. 2024).
- **DDiF** compresses *feature redundancy* by encoding data into neural fields that generate images on demand (Shin et al. 2025).

Although these methods improve storage efficiency, they still rely on 32-bit representations and incur high computational cost due to decoder networks. In contrast, our method uniquely targets *bit-level redundancy* through PTQ, enabling extremely low-bit (*e.g.*, 2-bit) storage. Table 1 summarizes recent PDC methods in terms of their redundancy type, compression strategy, approximate compression ratio (vs. 32-bit baseline), storage bitwidth, and computational overhead. Our method achieves the highest ratio via bit-level quantization while maintaining the lowest overhead.

Post-Training Quantization

Quantization reduces storage and computation by representing model weights or activations with fewer bits (Han, Mao, and Dally 2016; Esser et al. 2020). PTQ, in particular, is a widely adopted technique for efficient deployment on resource-constrained devices, since it avoids retraining (Zhang, Zhou, and Saab 2022; Zhihang Yuan 2022; Cai et al. 2020; Yao et al. 2023).

While PTQ has been extensively explored for compressing models, its application to dataset condensation has remained largely unexplored. To the best of our knowledge, this is the first work that applies post-training quantization to synthetic datasets in dataset condensation. Our method introduces a patch-based quantization framework with grouping and refinement, operating directly on 2-bit tensors without additional training or decoder models. This enables highly compressed, efficient, and distillation-free dataset storage and deployment.

Method

Figure 2 provides the overview of our proposed framework. In this section, we first provide background on DC and quan-

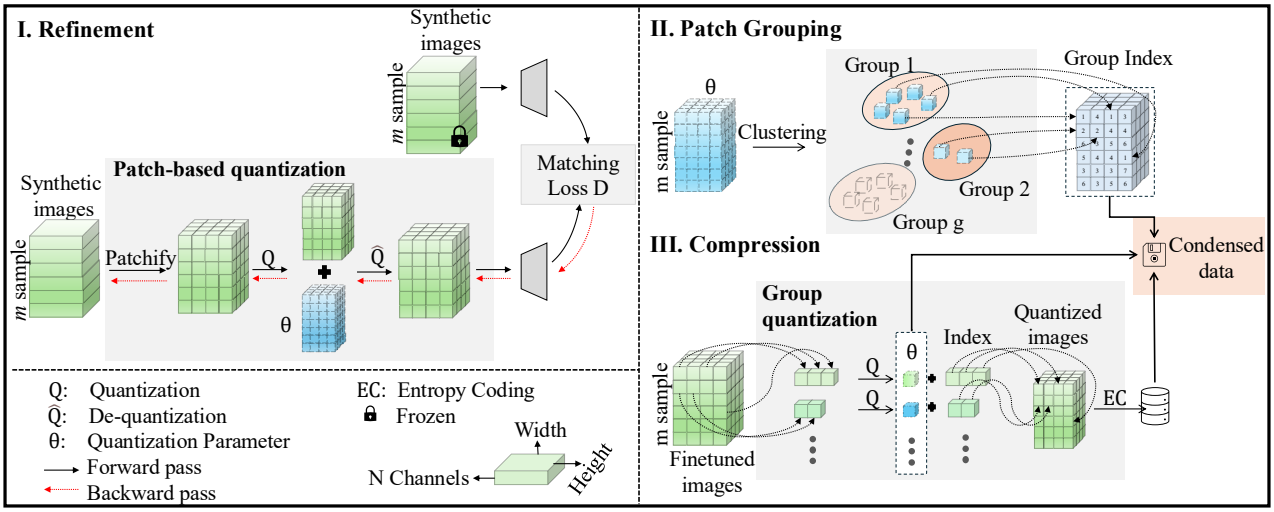


Figure 2: Overview of the proposed patch-based quantization framework for dataset condensation. (I) Synthetic images are first refined using a quantization-aware loss. (II) Patches are grouped based on quantization parameters via k-means clustering. (III) Each group is quantized with shared parameters, and the result is entropy encoded to produce the final compressed dataset.

tization. Then, we introduce our patch-based quantization. Next, we present our quantization grouping. Lastly, we describe the refinement module.

Preliminaries

Dataset Condensation. DC aims to synthesize a compact synthetic dataset $\mathcal{S} = \{(x^i, y^i) \mid i = 0, 1, \ldots, |\mathcal{S}| - 1\}$ from a larger dataset $\mathcal{T} = \{(x^i, y^i) \mid i = 0, 1, \ldots, |\mathcal{T}| - 1\}$, such using $\mathcal{S}$ for training enables the model to achieve a performance level comparable to that obtained with $\mathcal{T}$. Let $\mathcal{D}$ and Φ be the distance metric and the matching objective function. The condensation problem is formulated as the following optimization:

$$\mathcal{S}^* = \arg\min_{\mathcal{S}} \mathcal{D}\left(\Phi(\mathcal{S}), \Phi(\mathcal{T})\right), \tag{1}$$

where $\mathcal{S}^*$ is the optimal condensed set minimizing the discrepancy between the two objective functions.

Quantization. Quantization reduces the precision of model parameters by representing them with lower bit-width integers. A widely used method is *asymmetric quantization* (AQ) (Nagel et al. 2021), which employs two parameters: the *scale* and the *zero-point*. These enable the mapping of floating-point values to discrete integers. Let $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$ be an input tensor, and let $Q_{\text{min}} = 0$ and $Q_{\text{max}} = 2^b - 1$ denote the quantization bounds for a bit-width b. The scale factor α is computed as:

$$\alpha = \frac{\max(\mathbf{x}) - \min(\mathbf{x})}{Q_{\text{max}} - Q_{\text{min}}} \tag{2}$$

The zero-point z aligns the minimum value to the quantized range:

$$z = \left\lfloor Q_{\text{min}} - \frac{\min(\mathbf{x})}{\alpha} \right\rceil \tag{3}$$

where $\lfloor \cdot \rceil$ denotes rounding to the nearest integer. Given α and z, quantization and dequantization are defined as:

$$\mathbf{x}^q = \left\lfloor \frac{\mathbf{x}}{\alpha} + z \right\rceil, \quad \mathbf{x}^{\text{deq}} = (\mathbf{x}^q - z) \cdot \alpha. \tag{4}$$

Analysis of Whole-Image Quantization

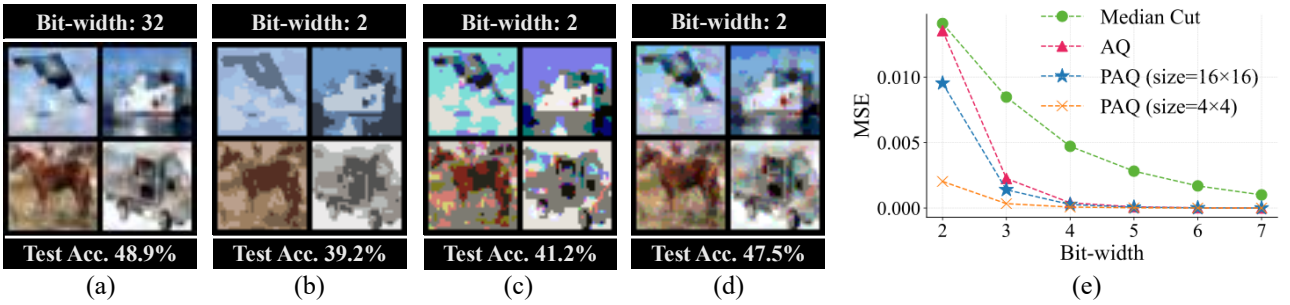
Whole-image quantization applies a single set of parameters (scale and zero-point) to the entire image. While simple and storage-efficient, this global approach suffers at extremely low bit-widths due to its inability to preserve fine-grained features. As a result, downstream performance degrades significantly.

We analyze this issue using two methods: (1) asymmetric quantization, and (2) Median Cut (Heckbert 1982), which partitions color space recursively along high-variance channels. We apply each method to synthetic images generated by DM (Zhao and Bilen 2022), then dequantize and train a 3-layer CNN for evaluation.

As shown in Fig. 3 (b,c), both whole-image quantization methods result in noticeable visual distortion and accuracy degradation compared to the original images (a). To quantify this, we measure the Mean Squared Error (MSE) between the feature embeddings of original and dequantized images. The results in Fig. 3 (e) validate that representational distortion increases as bit-width decreases. These findings suggest that whole-image quantization fails to maintain essential information needed for learning under aggressive compression.

Patch-based Quantization

To address the limitations of whole-image quantization and better align with the goals of dataset condensation, we introduce **patch-based asymmetric quantization** (PAQ). Unlike traditional approaches that apply a single global quantization



(a) (b) (c) (d) (e)

Figure 3: Visualization comparison of quantization strategies on synthetic images: (a) original images, (b) Median Cut (Heckbert 1982), (c) asymmetric quantization, and (d) our patch-based quantization. All quantized images use 2-bit precision. (e) shows distortion (MSE) versus bit-width across methods. Images are generated by DM (Zhao and Bilen 2022) on CIFAR-10.

to the entire image, PAQ operates at the patch level, enabling localized control of quantization and better preservation of fine-grained details.

Formally, given an image $\mathbf{x}$, we divide it into P non-overlapping patches $\{\mathbf{x}_i\}_{i=1}^{P}$, where each $\mathbf{x}_i \in \mathbb{R}^{h\times w\times C}$ represents a spatial region. We then perform per-patch quantization:

$$\mathbf{x}_i^q = Q(\mathbf{x}_i, \theta_i), \tag{5}$$

where $Q(\cdot, \theta_i)$ is the quantization function and $\theta_i = (\alpha_i, z_i)$ includes the scale and zero-point for patch i. This localized formulation allows each patch to adapt to its own distribution, minimizing information loss.

As shown in Fig. 3 (e), PAQ significantly reduces quantization-induced distortion across bit-widths compared to whole-image quantization. Notably, even at 2-bit precision, PAQ achieves competitive accuracy (47.5%) relative to full-precision data (48.9%), demonstrating its effectiveness in preserving task-relevant features under extreme compression.

Quantization-Aware Patch Grouping

While PAQ improves fidelity by assigning independent quantization parameters to each patch, it also increases storage cost due to the need to store θ_i for all patches. To mitigate this, we propose **group-aware quantization** (GAQ), which clusters similar patches to share quantization parameters.

We apply k-means clustering in the quantization parameter space to group patches with similar quantization behavior. Specifically, each patch $\mathbf{x}_i$ has an associated quantization parameter $\theta_i = (\alpha_i, z_i)$, where α_i and z_i denote the scale and zero-point, respectively. Let $\mathcal{C}_g$ denote the set of patches assigned to group g, and G be the total number of groups. Our objective is to minimize the intra-group variance of quantization parameters, encouraging patches within each group to share a common parameter:

$$\{\mathcal{C}_g^*, \theta_g^*\}_{g=1}^{G} = \underset{\{\mathcal{C}_g\},\{\theta_g\}}{\arg\min} \sum_{g=1}^{G} \sum_{\theta_i\in\mathcal{C}_g} \left\|\theta_i - \hat{\theta}_g\right\|^2, \tag{6}$$

$$\text{where } \hat{\theta}_g = \frac{1}{|\mathcal{C}_g|} \sum_{\theta_i\in\mathcal{C}_g} \theta_i.$$

where $\hat{\theta}_g$ represents the centroid (mean) of the quantization parameters in group g. This clustering step enables all patches in the same group to share a single quantization parameter θ_g, significantly reducing storage overhead while maintaining quantization fidelity.

Intra-group Recalibration. Rather than directly using the cluster centroids $\hat{\theta}_g$ as quantization parameters, we recalibrate θ_g based on the actual data distribution of patches within each group $\mathcal{C}_g$. Specifically, for each group g, we first concatenate all patches, $\mathbf{x}_g = \text{concat}(\{\mathbf{x}_i\}_{i\in\mathcal{C}_g})$, and then reshape the result into a one-dimensional tensor, $\mathbf{x}_g^{\text{flat}} = \text{reshape}(\mathbf{x}_g)$. We compute the group-specific quantization parameter θ_g by calibrating over $\mathbf{x}_g^{\text{flat}}$, and quantize the group as:

$$\mathbf{x}_g^q = Q(\mathbf{x}_g^{\text{flat}}, \theta_g), \tag{7}$$

where $Q(\cdot, \theta_g)$ is the quantization function. All patches in $\mathcal{C}_g$ are quantized using this shared parameter θ_g. This recalibration ensures that the shared quantization parameters accurately reflect the underlying patch distributions, thereby reducing quantization error and enhancing downstream model performance.

Refinement for Synthetic Images

Our goal is to refine synthetic images to mitigate the feature-space distortion caused by quantization. Prior studies have shown that quantization noise in inputs can significantly degrade the quality of extracted features (Sun et al. 2018; Zheng et al. 2016; Vasiljevic, Chakrabarti, and Shakhnarovich 2016).

Given a target bit-width b, we optimize a finetuned image $\mathbf{x}^{\text{ft}}$ such that its feature representation remains close to that of the original synthetic image, even after undergoing quantization and dequantization. Formally, we define:

$$(\mathbf{x}^{\text{ft}})^{\text{deq}} = \hat{Q}(Q(\mathbf{x}^{\text{ft}}, \theta), \theta), \tag{8}$$

where $Q(\cdot, \theta)$ and $\hat{Q}(\cdot, \theta)$ denote the quantization and dequantization functions parameterized by θ, respectively.

We extract features from both the original and the dequantized versions using a neural network $f(\cdot)$:

$$\mathbf{f} = f(\mathbf{x}), \quad \tilde{\mathbf{f}} = f((\mathbf{x}^{\text{ft}})^{\text{deq}}). \tag{9}$$

Dataset		CIFAR-10			CIFAR-100			Tiny ImageNet	
IPC		1	10	50	1	10	50	1	10
Distillation	DD	–	$36.8_{\pm 1.2}$	–	–	–	–	–	–
	DM	$26.0_{\pm 0.8}$	$48.9_{\pm 0.6}$	$63.0_{\pm 0.4}$	$11.4_{\pm 0.3}$	$29.7_{\pm 0.3}$	$43.6_{\pm 0.4}$	$3.9_{\pm 0.2}$	$12.9_{\pm 0.4}$
	DSA	$28.8_{\pm 0.7}$	$52.1_{\pm 0.5}$	$60.6_{\pm 0.5}$	$13.9_{\pm 0.3}$	$32.3_{\pm 0.3}$	$42.8_{\pm 0.4}$	–	–
	MTT	$46.3_{\pm 0.8}$	$65.3_{\pm 0.7}$	$71.6_{\pm 0.2}$	$24.3_{\pm 0.3}$	$40.1_{\pm 0.4}$	$47.7_{\pm 0.2}$	$8.8_{\pm 0.3}$	$23.2_{\pm 0.2}$
	DATM	$46.9_{\pm 0.5}$	$66.8_{\pm 0.2}$	$76.1_{\pm 0.3}$	$27.9_{\pm 0.2}$	$47.2_{\pm 0.4}$	$55.0_{\pm 0.2}$	$17.1_{\pm 0.3}$	$31.1_{\pm 0.3}$
Parameterization	IDC	$55.0_{\pm 0.4}$	$67.5_{\pm 0.5}$	$74.5_{\pm 0.1}$	–	–	–	–	–
	HaBa	$48.3_{\pm 0.8}$	$69.9_{\pm 0.4}$	$74.0_{\pm 0.2}$	$33.4_{\pm 0.4}$	$40.2_{\pm 0.2}$	$47.0_{\pm 0.2}$	–	–
	SPEED	$63.2_{\pm 0.1}$	$73.5_{\pm 0.2}$	$77.7_{\pm 0.4}$	$40.0_{\pm 0.4}$	$45.9_{\pm 0.3}$	$49.1_{\pm 0.2}$	$26.9_{\pm 0.3}$	$28.8_{\pm 0.2}$
	FreD	$60.6_{\pm 0.8}$	$70.3_{\pm 0.3}$	$75.8_{\pm 0.1}$	$34.6_{\pm 0.4}$	$42.7_{\pm 0.2}$	$47.8_{\pm 0.1}$	$19.2_{\pm 0.4}$	$24.2_{\pm 0.4}$
	Spectral	$68.5_{\pm 0.8}$	$73.4_{\pm 0.2}$	$75.2_{\pm 0.6}$	$36.5_{\pm 0.3}$	$46.1_{\pm 0.2}$	–	$21.3_{\pm 0.2}$	–
	AutoPalette	$58.6_{\pm 1.1}$	$74.3_{\pm 0.2}$	$79.4_{\pm 0.2}$	$38.0_{\pm 0.1}$	$52.6_{\pm 0.3}$	$53.3_{\pm 0.8}$	–	–
Quantization	DSA+Ours	$55.3_{\pm 0.4}$	$58.3_{\pm 0.5}$	$73.4_{\pm 0.4}$	$34.7_{\pm 0.3}$	$41.1_{\pm 0.5}$	–	–	–
	DM+Ours	$54.1_{\pm 0.5}$	$68.2_{\pm 0.4}$	$77.1_{\pm 0.3}$	$34.0_{\pm 0.3}$	$51.2_{\pm 0.3}$	–	$14.8_{\pm 0.5}$	$30.6_{\pm 0.3}$
	DATM+Ours	**$68.9_{\pm 0.4}$**	**$79.0_{\pm 0.3}$**	**$83.8_{\pm 0.2}$**	**$48.0_{\pm 0.3}$**	**$56.5_{\pm 0.2}$**	–	**$27.3_{\pm 0.5}$**	**$39.4_{\pm 0.4}$**
	Whole $\mathcal{T}$	$84.8_{\pm 0.1}$			$56.2_{\pm 0.2}$			$37.6_{\pm 0.4}$	

Table 2: Performance comparison on CIFAR-10, CIFAR-100, and Tiny ImageNet. Results are reported in *mean*$_{\pm std}$ format. Experiments on CIFAR-100 with IPC=50 are omitted, as this would require 1,000 images per class under a compression rate of 20, which exceeds the original 500 images per class.

To ensure that quantization does not introduce harmful feature drift, we minimize MSE between these feature representations:

$$\mathcal{L}_{\text{quant}} = \mathbb{E}_{\mathbf{x}\sim\mathcal{S}}\left[\left\|\mathbf{f} - \tilde{\mathbf{f}}\right\|_2^2\right], \tag{10}$$

where $\mathcal{S}$ denotes the set of synthetic images. The finetuned image $\mathbf{x}^{\text{ft}}$ is optimized to minimize this loss, thereby aligning its quantized feature representation with that of the original.

In practice, we explore three refinement strategies in our framework: (1) applying refinement only before group quantization, (2) only after group quantization, and (3) both before and after. We analyze the quantitative effects of each strategy in our ablation study and provide qualitative examples in the final part of the experimental section through visualization.

Storage Measurement

Under a fixed storage budget, our objective is to maximize the number of synthetic images per class. Starting from a baseline setting with IPC=m, we apply our compression framework to reduce the per-class footprint to IPC=n, thereby enabling a denser representation under the same storage constraint.

The total storage required to represent the synthetic data comprises three components: (i) **Group Indices** $\mathcal{G} = \{g_i\}_{i=1}^{P\times m}$, which indicate the group assignment for each patch, (ii) **Quantization Parameters** $\mathcal{Q} = \{\theta_g\}_{g=1}^{G}$, shared across patches within each group, and (iii) **Quantized Images** $\mathcal{X}^q = \{x_g^{\text{flat}}\}_{g=1}^{G}$, representing the flattened and quantized pixel values of each group. To further reduce the footprint of $\mathcal{X}^q$, we apply **Entropy Coding** (EC), which leverages the statistical redundancy in quantized values for additional compression.

Given a target storage budget corresponding to IPC=n, we ensure that the total compressed size does not exceed this constraint:

$$size(\mathcal{G}) + size(\mathcal{Q}) + size(\text{EC}(\mathcal{X}^q)) \leq size(\text{IPC}). \tag{11}$$

This formulation allows us to fit more quantized synthetic samples within the same memory budget, thereby improving the representational density of the condensed dataset.

Experiment

In this section, we evaluate the effectiveness of our proposed method in compressing synthetic images generated by various distillation techniques. We also conduct an ablation study to analyze the contribution of each component including group quantization, refinement, and entropy coding, to the overall performance.

Experimental Setting

We evaluate our method across diverse datasets, ranging from toy datasets like CIFAR (Krizhevsky and Hinton 2009) to large-scale, real-world datasets such as ImageNet subsets (Cazenavette et al. 2022), CC3M (Sharma et al. 2018), and Places365 (Zhou et al. 2017). As a plug-and-play framework, our method is applied to synthetic images generated by various distillation approaches, including DM (Zhao and Bilen 2022), DSA (Zhao and Bilen 2021), and DATM (Guo et al. 2024). Given a storage budget of IPC=m, we first generate synthetic images using existing methods, then compress them to IPC=n using our framework.

By default, we apply 2-bit quantization using non-overlapping patches of size 5×5. We perform a grid search to determine the maximum number of groups that satisfy the storage constraint. Prior to group quantization, synthetic images are refined for 500 iterations on CIFAR-10/100 and

Dataset	Nette	Woof	Fruit	Meow	Squawk	Yellow
MTT	$63.0_{\pm 1.3}$	$35.8_{\pm 1.8}$	$40.3_{\pm 1.3}$	$40.4_{\pm 2.2}$	$52.3_{\pm 1.0}$	$60.0_{\pm 1.5}$
DATM	$65.8_{\pm 2.0}$	$38.8_{\pm 1.6}$	$41.2_{\pm 1.4}$	$45.7_{\pm 1.8}$	$56.3_{\pm 1.5}$	$61.1_{\pm 1.8}$
HaBa	$64.7_{\pm 1.6}$	$38.6_{\pm 1.3}$	$42.5_{\pm 1.5}$	$42.9_{\pm 0.9}$	$56.8_{\pm 1.0}$	$63.0_{\pm 1.6}$
SPEED	$72.9_{\pm 1.5}$	$44.1_{\pm 1.4}$	$50.0_{\pm 0.8}$	$52.0_{\pm 1.3}$	$71.8_{\pm 1.3}$	$70.5_{\pm 1.5}$
FreD	$72.0_{\pm 0.8}$	$41.3_{\pm 1.2}$	$47.0_{\pm 1.1}$	$48.6_{\pm 0.4}$	$67.3_{\pm 0.8}$	$69.2_{\pm 0.6}$
AT	$73.2_{\pm 0.6}$	$44.3_{\pm 0.9}$	$48.4_{\pm 1.8}$	$53.6_{\pm 0.7}$	$68.0_{\pm 1.4}$	$72.0_{\pm 1.6}$
Ours	$\mathbf{81.1}_{\pm \mathbf{1.2}}$	$\mathbf{53.0}_{\pm \mathbf{0.8}}$	$\mathbf{56.6}_{\pm \mathbf{0.9}}$	$\mathbf{61.2}_{\pm \mathbf{0.9}}$	$\mathbf{80.6}_{\pm \mathbf{0.7}}$	$\mathbf{78.9}_{\pm \mathbf{0.9}}$

Table 3: Performance comparison on ImageNet subsets.

Dataset	CC3M		Places365 Standard	
IPC	1	10	1	10
Random	$2.9_{\pm 0.1}$	$7.0_{\pm 0.1}$	$0.9_{\pm 0.1}$	$3.9_{\pm 0.1}$
DM	$5.9_{\pm 0.2}$	$8.7_{\pm 0.1}$	$2.6_{\pm 0.1}$	$5.8_{\pm 0.1}$
IDC	$8.2_{\pm 0.3}$	$12.1_{\pm 0.2}$	$4.0_{\pm 0.1}$	$9.8_{\pm 0.1}$
DM+Ours	$\mathbf{9.9}_{\pm \mathbf{0.2}}$	$\mathbf{17.4}_{\pm \mathbf{0.1}}$	$\mathbf{7.7}_{\pm \mathbf{0.1}}$	$\mathbf{19.2}_{\pm \mathbf{0.1}}$
Whole $\mathcal{T}$	$46.7_{\pm 0.1}$		$40.4_{\pm 0.1}$	

Table 4: Performance comparison on real-world datasets.

2000 iterations on ImageNet subsets. For evaluation, models are trained on the compressed synthetic datasets and tested on the original test sets.

Experimental Results

Results on small-scale datasets. We compare our quantization-based parameterization method against various PDC baselines, including Factorization (Liu et al. 2022) (HaBa), Coding Matrices (Wei et al. 2023) (SPEED), Frequency Domain (Shin, Shin, and Moon 2023) (FreD), Spectral Domain (Yang et al. 2025) (Spectral), and Color Redundancy Reduction (Yuan et al. 2024) (AutoPalette).

As shown in Table 2, our framework significantly improves the performance of DM and DSA. For instance, at IPC=1, DM achieves 26% accuracy, whereas DM+Ours achieves 54.1%, a relative gain of over 100%. A similar trend is observed for DSA. When applied to the latest distillation method DATM, our framework achieves state-of-the-art performance across all IPC levels. Although AutoPalette (Yuan et al. 2024) reduces color redundancy via an additional network, it incurs higher computational cost. Despite sharing the same distillation base (DATM), our method consistently outperforms AutoPalette, demonstrating both higher efficiency and accuracy.

Results on mid- and large-scale datasets. Following (Cazenavette et al. 2022), we evaluate our method on high-resolution ImageNet subsets. As shown in Table 3, integrating our framework into DATM consistently yields the best performance across all subsets, with gains ranging from 6.9% (Yellow) to 12.6% (Squawk). Furthermore, when evaluating our method on CC3M and Places365, we observe the highest test accuracy in all cases (see Table 4). These results demonstrate the scalability of our framework to more complex domains.

Method	IPC=10	IPC=20
Random	42.6	57.0
Herding	56.2	72.9
DM	69.1	77.2
IDC	82.9	86.6
DDiF	90.5	92.7
DM+Ours	**93.1**	**95.0**
Whole $\mathcal{T}$	93.4	

Table 5: Performance comparison on Audio domain.

$\mathcal{L}$	Method	MNet	SNet
DC	IDC	78.7	79.9
	DDiF	87.1	89.6
	Ours	**93.7**	**91.1**
DM	IDC	85.6	85.3
	DDiF	88.4	93.1
	Ours	**93.9**	**93.1**
Whole $\mathcal{T}$		91.6	98.3

Table 6: Performance comparison on 3D voxel domain at IPC=1.

Method	Network			
	ConvNet	AlexNet	VGG11	ResNet18
DC	$44.9_{\pm 0.5}$	$22.4_{\pm 1.4}$	$35.9_{\pm 0.7}$	$18.4_{\pm 0.4}$
MTT	$65.3_{\pm 0.7}$	$34.2_{\pm 0.6}$	$50.3_{\pm 0.8}$	$46.6_{\pm 0.6}$
DATM	$66.8_{\pm 0.2}$	$32.7_{\pm 3.8}$	$38.8_{\pm 2.1}$	$51.4_{\pm 1.7}$
FRePo	$65.5_{\pm 0.4}$	$61.9_{\pm 0.7}$	$59.4_{\pm 0.7}$	$58.1_{\pm 0.6}$
Spectral	$73.4_{\pm 0.2}$	$71.4_{\pm 0.3}$	$67.8_{\pm 0.2}$	$64.9_{\pm 1.3}$
DATM+Ours	$\mathbf{79.0}_{\pm \mathbf{0.3}}$	$\mathbf{74.7}_{\pm \mathbf{0.5}}$	$\mathbf{74.7}_{\pm \mathbf{0.3}}$	$\mathbf{77.5}_{\pm \mathbf{0.3}}$

Table 7: Cross-architecture performance comparison between our framework and other methods at IPC=10.

Generalization to other modalities. To demonstrate versatility, we apply our framework to audio and 3D data domains following the experiment setup in (Shin et al. 2025). As shown in Table 5 and Table 6, our method outperforms recent approaches such as DDiF (Shin et al. 2025), highlighting its cross-modal robustness.

Cross-architecture validation. We further assess generalization across architectures. Using synthetic images generated by DATM, we apply our method and evaluate on AlexNet (Krizhevsky, Sutskever, and Hinton 2012), VGG (Simonyan and Zisserman 2015), and ResNet (He et al. 2015). As shown in Table 7, our method demonstrates superior generalization performance compared to previous approaches.

GAQ	Refinement	EC	CIFAR-10	Nette
✗	✗	✗	$71.8_{\pm 0.3}$	$75.2_{\pm 0.8}$
✓	✗	✗	$76.1_{\pm 0.3}$	$76.5_{\pm 1.1}$
✓	✓	✗	$77.2_{\pm 0.3}$	$77.2_{\pm 1.0}$
✓	✓	✓	$\mathbf{79.0}_{\pm \mathbf{0.3}}$	$\mathbf{81.1}_{\pm \mathbf{1.2}}$

Table 8: Effect of each component on final performance. The first row shows the baseline using global asymmetric quantization. IPC is set to 10 for both CIFAR-10 and Nette dataset.

Refinement before GAQ	✓	✗	✓
Refinement after GAQ	✗	✓	✓
Test Accuracy (%)	$68.9_{\pm 0.4}$	$68.7_{\pm 0.5}$	$68.9_{\pm 0.3}$

Table 9: Test accuracy on CIFAR-10 at IPC=1 with refinement applied at different stages.

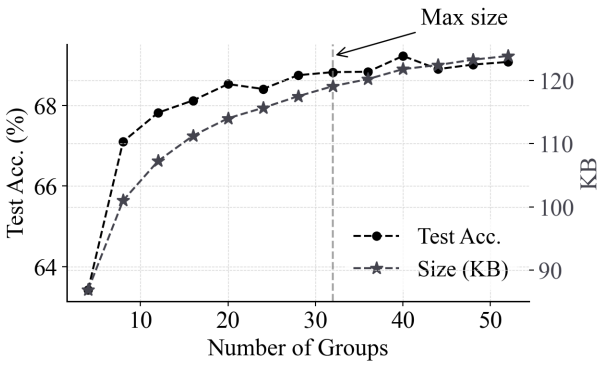


Figure 4: Measured test accuracy (%) and storage across different numbers of groups. The vertical dotted line marks the maximum allowable storage under the given budget.

Ablation Study

We conduct an ablation study to analyze the effect of each component in our framework on the final performance, as well as the impact of the number of groups. By default, we adopt DATM to generate synthetic images.

Effect of Each Component. As shown in Table 8, applying asymmetric quantization (AQ) alone achieves 71.8% test accuracy on CIFAR-10, outperforming the base DATM method (66.8%) at IPC=10. Adding group-aware quantization (GAQ) further improves accuracy to 76.1%, and incorporating the refinement module increases it to 77.2%. Finally, applying entropy coding allows more samples to be stored within the same budget, leading to a final performance of 79%.

Ablation on Refinement Timing. As shown in Table 9, applying refinement before group quantization yields better performance than applying it afterward. This is likely because group quantization assigns patches to groups based on their quantization parameters, which are more accurately estimated when the image is refined beforehand. In contrast, applying group quantization first may lead to suboptimal grouping due to uncorrected quantization noise. Notably, performing refinement both before and after group quantization does not lead to further improvement, suggesting diminishing returns from post-group refinement.

Ablation on the Number of Groups. Figure 4 shows how the number of groups (G in Eq. (6)) affects performance. We observe that increasing the number of groups consistently improves test accuracy, while staying within the storage budget. Based on this trend, we choose the maximum number of groups that satisfies the storage constraint.

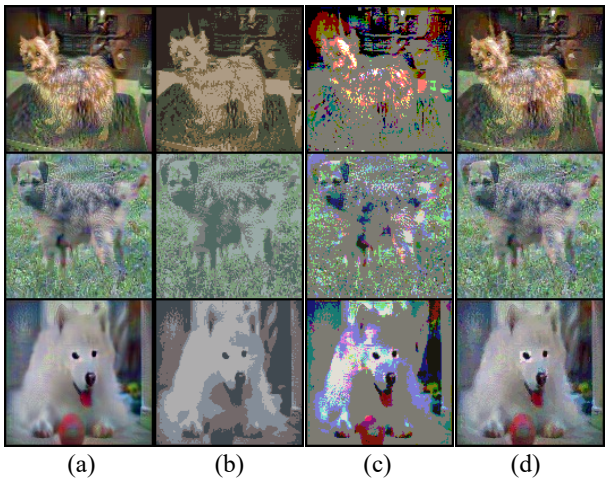


Figure 5: Visualization of (a) original synthetic images and their quantized versions using (b) Median Cut, (c) Asymmetric Quantization (AQ), and (d) Group-Aware Quantization (GAQ). While Median Cut preserves texture but distorts color, and AQ preserves color but loses fine details, GAQ achieves a better balance between texture and color fidelity.

Visualization of Synthetic Images. We visualize quantized synthetic images generated by MTT (Cazenavette et al. 2022) on ImageNet Woof. As shown in Figure 5, the Median Cut method retains texture but alters color severely, while AQ preserves color at the cost of heavy texture distortion. In contrast, our group-based quantization strikes a better balance, maintaining both texture and color fidelity. Additional examples are provided in the supplementary material.

Conclusion

We propose a novel quantization framework for dataset condensation that achieves substantial storage reduction while preserving downstream performance. Unlike prior work that focuses solely on synthetic image generation, our method explicitly targets bit-level redundancy through patch-based quantization, group-wise parameter sharing, and lightweight feature refinement. This combination enables effective compression even at extremely low bit-widths. Extensive experiments across diverse distillation methods and datasets validate that our approach consistently outperforms existing parameterized condensation baselines under tight storage constraints.

Limitation and Future Work. Our framework is currently designed for condensation pipelines that focus on visual data and hard labels. It does not yet support methods that utilize soft labels. Future work may explore label-aware quantization. In addition, our method is tailored for CNN-based architectures operating on spatial patches, and has not been extended to transformers. Designing quantization strategies that operate on token embeddings presents a promising direction for transformer-based models.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2025-24534076) and (No. RS-2025-02216217), and by the IITP (Institute of Information & Communications Technology Planning & Evaluation) - ITRC (Information Technology Research Center) grant funded by the Korea government (Ministry of Science and ICT) (IITP-2025-RS-2024-00438239)

References

Cai, Y.; Yao, Z.; Dong, Z.; Gholami, A.; Mahoney, M. W.; and Keutzer, K. 2020. Zeroq: A novel zero shot quantization framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13169–13178.

Cazenavette, G.; Wang, T.; Torralba, A.; Efros, A. A.; and Zhu, J.-Y. 2022. Dataset Distillation by Matching Training Trajectories. In *CVPR*.

Chai, Z.; Gao, Z.; Lin, Y.; Zhao, C.; Yu, X.; and Xie, Z. 2024. Adaptive Backdoor Attacks Against Dataset Distillation for Federated Learning. In *ICC*, 4614–4619.

Cui, J.; Wang, R.; Si, S.; and Hsieh, C.-J. 2023. Scaling up dataset distillation to imagenet-1k with constant memory. In *International Conference on Machine Learning*, 6565–6590. PMLR.

Dong, T.; Zhao, B.; and Lyu, L. 2022. Privacy for Free: How does Dataset Condensation Help Privacy? In *ICML*, volume 162, 5378–5396. PMLR.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 3749–3758.

Esser, S. K.; McKinstry, J. L.; Bablani, D.; Appuswamy, R.; and Modha, D. S. 2020. LEARNED STEP SIZE QUANTIZATION. In *International Conference on Learning Representations*.

Gu, J.; Wang, K.; Jiang, W.; and You, Y. 2024. Summarizing Stream Data for Memory-Constrained Online Continual Learning. arXiv:2305.16645.

Guo, Z.; Wang, K.; Cazenavette, G.; LI, H.; Zhang, K.; and You, Y. 2024. Towards Lossless Dataset Distillation via Difficulty-Aligned Trajectory Matching. In *The Twelfth International Conference on Learning Representations*.

Han, S.; Mao, H.; and Dally, W. J. 2016. Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding. *International Conference on Learning Representations (ICLR)*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Deep Residual Learning for Image Recognition. *CoRR*, abs/1512.03385.

Heckbert, P. 1982. Color image quantization for frame buffer display. In *ACM*.

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

Krizhevsky, A.; and Hinton, G. 2009. Learning multiple layers of features from tiny images.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In Pereira, F.; Burges, C.; Bottou, L.; and Weinberger, K., eds., *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc.

Liu, S.; Wang, K.; Yang, X.; Ye, J.; and Wang, X. 2022. Dataset Distillation via Factorization. In *NeurIPS*.

Nagel, M.; Fournarakis, M.; Amjad, R. A.; Bondarenko, Y.; van Baalen, M.; and Blankevoort, T. 2021. A White Paper on Neural Network Quantization. *CoRR*, abs/2106.08295.

Redmon, J.; Divvala, S.; Girshick, R.; and Farhadi, A. 2016. You Only Look Once: Unified, Real-Time Object Detection. In *CVPR*.

Sajedi, A.; Khaki, S.; Amjadian, E.; Liu, L. Z.; Lawryshyn, Y. A.; and Plataniotis, K. N. 2023. DataDAM: Efficient Dataset Distillation with Attention Matching. In *ICCV*, 17097–17107.

Sharma, P.; Ding, N.; Goodman, S.; and Soricut, R. 2018. Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning. In *Proceedings of ACL*.

Shin, D.; Bae, H.; Sim, G.; Kang, W.; and chul Moon, I. 2025. Distilling Dataset into Neural Field. In *The Thirteenth International Conference on Learning Representations*.

Shin, D.; Shin, S.; and Moon, I.-C. 2023. Frequency Domain-Based Dataset Distillation. In *NeurIPS*.

Simonyan, K.; and Zisserman, A. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv:1409.1556.

Sun, Z.; Ozay, M.; Zhang, Y.; Liu, X.; and Okatani, T. 2018. Feature Quantization for Defending Against Distortion of Images. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7957–7966.

Vasiljevic, I.; Chakrabarti, A.; and Shakhnarovich, G. 2016. Examining the Impact of Blur on Recognition by Convolutional Networks. *CoRR*, abs/1611.05760.

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 *CVPR*, 12186–12195.

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

Wei, X.; Cao, A.; Yang, F.; and Ma, Z. 2023. Sparse Parameterization for Epitomic Dataset Distillation. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Yang, E.; Shen, L.; Wang, Z.; Liu, T.; and Guo, G. 2023. An Efficient Dataset Condensation Plugin and Its Application to Continual Learning. In *NeurIPS*.

Yang, S.; Cheng, S.; Hong, M.; Fan, H.; Wei, X.; and Liu, S. 2025. Neural Spectral Decomposition for Dataset Distillation. In Leonardis, A.; Ricci, E.; Roth, S.; Russakovsky, O.; Sattler, T.; and Varol, G., eds., *Computer Vision – ECCV 2024*, 275–290. Cham: Springer Nature Switzerland.

Yao, Z.; Wu, X.; Li, C.; Youn, S.; and He, Y. 2023. ZeroQuant-V2: Exploring Post-training Quantization in LLMs from Comprehensive Study to Low Rank Compensation. arXiv:2303.08302.

Yuan, B.; Wang, Z.; Baktashmotlagh, M.; Luo, Y.; and Huang, Z. 2024. Color-Oriented Redundancy Reduction in Dataset Distillation. In *NeurIPS*.

Zhang, H.; Su, S.; Zhu, Y.; Sun, J.; and Zhang, Y. 2024. GSDD: Generative Space Dataset Distillation for Image Super-resolution. In *AAAI*, 7069–7077.

Zhang, J.; Zhou, Y.; and Saab, R. 2022. Post-training Quantization for Neural Networks with Provable Guarantees. *CoRR*, abs/2201.11113.

Zhao, B.; and Bilen, H. 2021. Dataset Condensation with Differentiable Siamese Augmentation. In *ICML*, volume 139, 12674–12685. PMLR.

Zhao, B.; and Bilen, H. 2022. Dataset Condensation with Distribution Matching. arXiv:2110.04181.

Zhao, B.; Mopuri, K. R.; and Bilen, H. 2021. Dataset Condensation with Gradient Matching. In *ICLR*.

Zhao, G.; Li, G.; Qin, Y.; and Yu, Y. 2023. Improved Distribution Matching for Dataset Condensation. In *CVPR*, 7856–7865.

Zheng, S.; Song, Y.; Leung, T.; and Goodfellow, I. 2016. Improving the Robustness of Deep Neural Networks via Stability Training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhihang Yuan, Y. C. Q. W. G. S., Chenhao Xue. 2022. PTQ4ViT: Post-Training Quantization Framework for Vision Transformers. *arXiv preprint arXiv:2111.12293*.

Zhou, B.; Lapedriza, A.; Khosla, A.; Oliva, A.; and Torralba, A. 2017. Places: A 10 million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.