IntraQ: Learning Synthetic Images with Intra-Class Heterogeneity for Zero-Shot Network Quantization

Yunshan Zhong[1,2], Mingbao Lin[2], Gongrui Nan[2], Jianzhuang Liu[3],
Baochang Zhang[4], Yonghong Tian[5,6], Rongrong Ji[1,2,6*]
[1]Institute of Artificial Intelligence, Xiamen University
[2]MAC Lab, School of Informatics, Xiamen University [3]Noah's Ark Lab, Huawei
[4]Beihang University [5]Peking University [6]Peng Cheng Laboratory

Abstract

Learning to synthesize data has emerged as a promising direction in zero-shot quantization (ZSQ), which represents neural networks by low-bit integer without accessing any of the real data. In this paper, we observe an interesting phenomenon of intra-class heterogeneity in real data and show that existing methods fail to retain this property in their synthetic images, which causes a limited performance increase. To address this issue, we propose a novel zero-shot quantization method referred to as IntraQ. First, we propose a local object reinforcement that locates the target objects at different scales and positions of the synthetic images. Second, we introduce a marginal distance constraint to form class-related features distributed in a coarse area. Lastly, we devise a soft inception loss which injects a soft prior label to prevent the synthetic images from being over-fitting to a fixed object. Our IntraQ is demonstrated to well retain the intra-class heterogeneity in the synthetic images and also observed to perform state-of-the-art. For example, compared to the advanced ZSQ, our IntraQ obtains 9.17% increase of the top-1 accuracy on ImageNet when all layers of MobileNetV1 are quantized to 4-bit. Code is at https://github.com/zysxmu/IntraQ.

1. Introduction

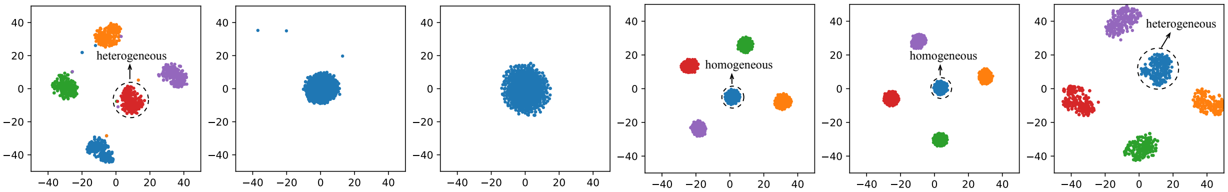
The increasing demands in computing power and memory footprint of deep neural networks (DNNs) raise a challenging application problem on edge computing devices such as smart phones or wearable gadgets, in which the limited hardware resource fails to support the highly complex DNNs. A variety of methods [12, 13, 20, 25] have been investigated to reduce the model complexity. Network quantization, which represents the floating-point parameters and activations within the networks by low-bit integers, stands out among these methods for its significant memory reduction and more efficient integer operations.

Most existing methods explore quantization-aware training (QAT) that builds a quantizer on the premise of accessing the original complete training dataset [2, 8, 43]. In [18, 23, 47], QAT is demonstrated to be comparable or even better than its floating-point counterpart since the weights could be adjusted to fit the quantization operations given the access to sufficient training data [40]. However, the drawbacks also stem from its reliance on training data. Specifically, in real-world cases, the original training data is sometimes prohibitive due to deteriorating privacy and security problems. For example, people may not wish their medical records to be revealed to others, and business material is not expected to be transmitted via the internet. As such, QAT is no longer applicable. Though recent studies on post-training quantization (PTQ) [24, 30, 40] directly quantizes DNNs using a small portion of original data, for cases such as MLaas (*e.g.*, Amazon AWS and Google Cloud), it may be impossible to reach any of the training data from users [3].

Fortunately, the research community recently has proposed zero-shot quantization (ZSQ) to quantize models without accessing real data. Existing studies on ZSQ can be categorized into two groups. The first group calibrates parameters without the involvement of any data. For example, DFQ [31] utilizes the shift and scale parameters $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ stored in the batch normalization layers of the full-precision model to compute the expected biased error on the output. Nevertheless, a simple calibration of parameters results in severe performance degradation in ultra-low precision. For instance, only 0.10% top-1 accuracy of DFQ on ImageNet [36] is reported in the appendix of [41] if quantizing ResNet-18 to 4-bit.

The second group performs quantization by exploiting synthetic fake images. The involvement of fake images facilitates the training of quantized networks which are demonstrated to be superior in performance [11, 41, 46].

[*] Corresponding Author: rrji@xmu.edu.cn



(a) Real data (b) ZeroQ (c) DSG (d) ZeroQ+IL (e) DSG+IL (f) **IntraQ** (Ours)

Figure 1. Feature visualization using t-SNE [39]. We randomly sample 1,000 synthetic/real images consisting of 5 classes with 200 images per class. For ZeroQ and DSG, label information is unavailable. The features are extracted from a pre-trained ResNet-18.

An intuitive solution is to deploy a generator to synthesize training data [4, 28, 41]. However, these generator-based methods suffer a heavy overhead on computation resources since the generator has to be trained from scratch for different bit-width settings. On the contrary, many studies such as ZeroQ [3] and DSG [46] formulate the data synthesis as an optimization problem where a random input data drawn from the standard Gaussian distribution is iteratively updated to fit the real-data distribution. This research line leads to a resource-friendly quantization as the synthetic images can be reused to calibrate or fine-tune networks in different bit widths. However, a non-ignorable quality gap in synthetic images still remains when comparing the feature visualization of ZeroQ (Fig. 1b) and DSG (Fig. 1c) with the real data (Fig. 1a) since traditional Gaussian synthesis is towards fitting the whole dataset while ignoring a subtler class-wise decision boundary. Thus, the quantized models often bear large performance drops (see Sec. 3.1.2).

To ensure class-wise discrimination in the fake images, we apply the popular inception loss [9, 44] to ZeroQ and DSG, which first chooses an arbitrary label, and then performs optimization to generate label-oriented images. As a result, we observe more class-wise separable distributions of synthetic data (Fig. 1d and Fig. 1e). This demonstrates the importance of injecting prior class information in synthetic data. Nevertheless, we observe that the synthetic data with the inception loss fail to capture the intra-class heterogeneity. Specifically, images from the same class often contain different contents; thus features from the same class of real data scatter a lot as shown in in Fig. 1a. On the contrary, those in Fig. 1d and Fig. 1e are in a dense concentration, which indicates the synthetic images of the same class are mostly homogeneous. Consequently, the quantized model fine-tuned with these synthetic data fails to generalize well to the real-world test dataset featuring heterogeneity.

To retain the intra-class heterogeneity, in this paper, we propose a novel zero-shot quantization method, termed IntraQ. Motivated by the fact that the objects of interest benefiting the model learning are not always at the same scale or position in the images, we propose a local object reinforcement by randomly cropping a local patch from the synthetic image to locate the target objects, which mitigates synthesizing homogeneous images. Apart from heterogeneous images, we also propose to retain the intra-class heterogeneity in their feature space. This is accomplished by introducing a marginal distance constraint to not only form class-related features but also avoid learning features concentrated on a dense area. In contrast to the traditional inception loss with one-hot label, we further devise a soft inception loss which injects a soft prior label to excavate images with more complex scenes and prevent the synthetic images from being overfitting to a fixed object. With the above three innovative solutions, the intra-class heterogeneity is well preserved in our synthetic images as shown in Fig. 1f and significant performance improvements are observed when using only 5,120 synthetic images to fine-tune the quantized models. For instance, our IntraQ achieves 51.36% top-1 accuracy on ImageNet when quantizing MobileNetV1 to 4-bit, leading to a increase of 9.17% when compared with the advanced DSG [46] equipped with the traditional inception loss [9].

2. Related Work

2.1. Data-Driven Quantization

Both QAT and PTQ require real data to complete quantization. With abundant training images, existing QAT methods focus on designing quantizers [6,17,23], training strategies [22,47], dynamic quantization [16,38,45], binary networks [26,29,34,35], approximate gradients [8,42], *etc*. On the contrary, PTQ is limited to accessing a very small portion of training data [1, 7, 24, 27, 30, 40]. Banner *et al*. [1] combined analytical clipping, per-channel bit allocation, and bias-correction to form a 4-bit post-training method. AdaRound [30] shows that the rounding-to-nearest is not the optimal rounding function and formulates the rounding as a layer-wise quadratic unconstrained binary problem. In [27], a linear combination of multiple low-bit vectors is used to approximate a full-precision weight vector. A mixed-precision network is constructed upon a quantization error-based greedy selection to adaptively decide the number of low-bit vectors. Based on a theoretical study of the second-order loss and empirical evidence, Li *et al*. [24] proposed a block reconstruction to regain the accuracy.

2.2. Zero-Shot Quantization

ZSQ performs network quantization without accessing any real data. To this end, DFQ [31] focuses on calibrating network parameters by utilizing the scale-equivariance property. To fix inherent bias [1] without data, the shift parameter $\boldsymbol{\beta}$ and scale parameter $\boldsymbol{\gamma}$ in the BN layers are used to calculate the expected biased error on the outputs. Another group concentrates on synthesizing fake images for better peformance. GDFQ [41] integrates the BNS alignment loss and inception loss to train a generator for generating label-oriented images. To diversify synthetic images, DQAKD [4] trains the generator in an adversarial manner. ZAQ [28] also adversarially trains the generator with a novel two-level modeling strategy to measure the discrepancy. In addition to the generator, the data synthesis can also be realized by optimizing the Gaussian noise. By regarding batch normalization statistics (BNS) (*i.e.* running mean $\boldsymbol{\mu}$ and running variance $\boldsymbol{\sigma}^2$) as the distribution indicators, ZeroQ [3] optimizes the Gaussian noise until the mean and variance of synthetic data can match the BNS in the pre-trained network. DSG [46] relaxes the BNS alignment loss to prevent synthetic images from over-fitting, and randomly enlarges the loss term for each sample in backward propagation. Inspired by VAE [5], GZNQ [11] regards the synthetic images as optimizable parameters and introduces ensembling to model hard samples. By approximating BNS, [14] estimates the fake data to determine activation ranges.

3. Methodology

3.1. Preliminaries

3.1.1 Quantizer

Following the settings of [41], we use the asymmetric uniform quantizer to implement network quantization. Denoting $\boldsymbol{x}$ as weights/activations, l and u as the lower bound and upper bound of $\boldsymbol{x}$, we can obtain the quantized integer $\boldsymbol{q}$ as:

$$\boldsymbol{q} = round\Big(\frac{clip(\boldsymbol{x}, l, u)}{s}\Big), \tag{1}$$

where $clip(\boldsymbol{x}, l, u) = min(max(\boldsymbol{x}, l), u)$ and $round(\cdot)$ rounds its input to the nearest integer. $s = \frac{u-l}{2^b-1}$ is the scaling factor that projects a floating-point number to a fixed-point integer and b is the bit-width. The corresponding de-quantized value $\bar{\boldsymbol{x}}$ can be calculated as:

$$\bar{\boldsymbol{x}} = \boldsymbol{q} \cdot s. \tag{2}$$

For activations and weights, we use layer-wise quantizer and channel-wise quantizer respectively.

3.1.2 Data Synthesis

ZSQ receives popularity mostly due to its evasion of accessing real data. However, its limited performance also results from this limitation. By making full use of the pre-trained full-precision model F to generate fake images, data synthesis has garnered more attention recently since the involvement of fake images greatly facilitates the training of quantized networks. One basic principle in data synthesis is to fit the real-data distribution, which is explored by the BNS alignment loss that aligns the batch normalization statistics (BNS) in many existing studies [3, 41, 46] as:

$$\mathcal{L}_{\text{BNS}}(\tilde{\boldsymbol{I}}) = \sum_{l=1}^{L} \|\boldsymbol{\mu}'_l(\tilde{\boldsymbol{I}}) - \boldsymbol{\mu}^F_l\|^2 + \|\boldsymbol{\sigma}'_l(\tilde{\boldsymbol{I}}) - \boldsymbol{\sigma}^F_l\|^2_2, \tag{3}$$

where $\boldsymbol{\mu}^F_l$ and $\boldsymbol{\sigma}^F_l$ are the running mean and variance stored in the l-th BN layer of the pre-trained full-precison network F, and $\boldsymbol{\mu}'_l(\tilde{\boldsymbol{I}})$ and $\boldsymbol{\sigma}'_l(\tilde{\boldsymbol{I}})$ denote the mean and variance of synthetic image batch $\tilde{\boldsymbol{I}}$ in the l-th layer of F, respectively.

However, we observe that similar mean and variance do not indicate an identical data distribution. As shown in Fig. 1b and Fig. 1c, the distributions of synthetic fake images from ZeroQ [3] and DSG [46] differ a lot from that of the real data in Fig. 1a. Particularly, a subtler class-wise distribution is overlooked since the synthesis is to fit the mean and variance of the whole dataset without any label information. The poor quality of synthetic data also results in inferior performance of 60.68% for ZeroQ and 60.12% for DSG on ImageNet when all layers of ResNet-18 are quantized to 4-bit as experimentally shown in Tab. 1.

Fortunately, the inception loss [9], which first chooses an arbitrary label y as a prior classification knowledge and then performs optimization to generate these label-oriented images, might be a potential method to solve this problem. It can be formulated as:

$$\mathcal{L}_{\text{IL}}(\tilde{\boldsymbol{I}}) = ce\big(F(\tilde{\boldsymbol{I}}), y\big), \tag{4}$$

where $ce(\cdot, \cdot)$ represents the cross entropy and $F(\cdot)$ returns a probability distribution, *i.e.*, the output of the softmax layer. Note that F is fixed and the gradient will be backwarded to optimize the synthetic images $\tilde{\boldsymbol{I}}$ for fitting the distribution of real images from class y. As shown in Fig. 1d and Fig. 1e. the distributions of fake data from ZeroQ and DSG become class-wise discriminative, and are more close to that of the real data after integrating the inception loss. Consequently, the performances of ZeroQ and DSG respectively increase to 63.38% and 63.11% in Tab. 1, demonstrating the efficacy of incorporating prior class information in synthetic images.

3.2. Our Insights

Though existing ZSQ methods benefit from the inception loss, the performance gain is limited if compared with 67.89% of quantized ResNet-18 fine-tuned on real training data in Tab. 1. To dive into a deeper analysis, when looking back into Fig. 1, we observe that though class-wise discriminative, the synthetic data with inception loss does not well

Method	Avg. of intra-class cosine distances	Acc. (%)
full-precision	-	71.49
Real data	0.44	67.89
ZeroQ	-	60.68
DSG	-	60.12
ZeroQ+IL	0.17	63.38
DSG+IL	0.19	63.11
IntraQ (Ours)	0.42	**66.47**

Table 1. Top-1 accuracy of 4-bit ResNet-18 fine-tuned on 5,120 fake/real images and average of intra-class cosine distances. The "IL" is short for inception loss.

capture the intra-class heterogeneity. Images, even from the same class, often contain different contents, and thus features from the same class in Fig. 1a scatter a lot. On the contrary, these in Fig. 1d and Fig. 1e are in a dense concentration, which indicates that the synthetic images from the same class are mostly homogeneous. Quantized models fine-tuned on these homogeneous fake images fail to well generalize to the real-world test dataset featuring heterogeneity. Thus, the performance gains becomes limited.

To quantitatively measure the intra-class heterogeneity, we feed the synthetic images to the pre-trained full-precision ResNet-18 to derive their feature vectors and then calculate the cosine distances among feature data from the same class. Tab. 1 displays the average cosine distance of the synthetic images. It is easy to understand that the intra-class heterogeneity can be well reflected by the cosine distance (ranging from 0 to 2). Note that the quantitative results for ZeroQ and DSG are not presented since the synthetic images are unlabeled when the inception loss is not applied. From Tab. 1, the average distance for real data is 0.44, which denotes a high degree of intra-class heterogeneity in real data. This statistical result conforms with the scattered intra-class visualization in Fig. 1a. However, the average distances for ZeroQ and DSG with the inception loss are only 0.17 and 0.19, less than half of the real data. Consequently, synthetic images from the same class tend to be densely distributed in a small area as shown in Fig. 1d and Fig. 1e. Thus, the inception loss fails to retain the intra-class heterogeneity, which however, if well addressed, might be a promise of further boosting the performance of ZSQ.

3.3. Our Solutions

In this section, we introduce our proposed IntraQ to learn synthetic images with intra-class heterogeneity. As shown in Fig. 2, the core contributions of our IntraQ are three folds: a local object reinforcement, a marginal distance constraint, and a soft inception loss.

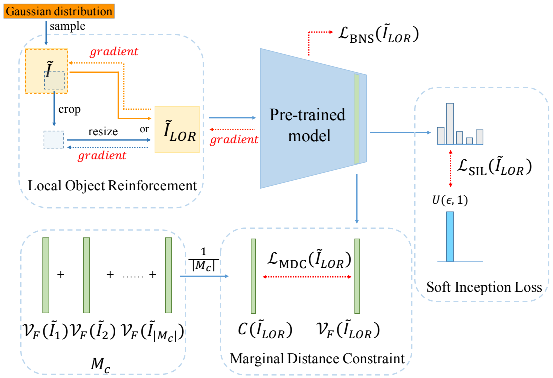

Gaussian distribution
sample
gradient
$\tilde{I}$
crop
resize
or
gradient
$\tilde{I}_{LOR}$
gradient
Local Object Reinforcement
Pre-trained model
$\mathcal{L}_{\text{BNS}}(\tilde{I}_{LOR})$
$\mathcal{L}_{\text{SIL}}(\tilde{I}_{LOR})$
$U(\epsilon, 1)$
Soft Inception Loss
+
+ ...... +
$\mathcal{V}_F(\tilde{I}_1)$ $\mathcal{V}_F(\tilde{I}_2)$ $\mathcal{V}_F(\tilde{I}_{|M_c|})$
M_c
$\frac{1}{|M_c|}$
$\mathcal{L}_{\text{MDC}}(\tilde{I}_{LOR})$
$\mathcal{C}(\tilde{I}_{LOR})$ $\mathcal{V}_F(\tilde{I}_{LOR})$
Marginal Distance Constraint

Figure 2. The framework of our IntraQ. The local object reinforcement locates objects at different scales and positions of the synthetic image $\tilde{\boldsymbol{I}}$. The marginal distance constraint forms heterogeneous intra-class features. The soft inception injects soft label information to learn complex scenes in the synthetic images.

3.3.1 Local Object Reinforcement

Our first step to retain the intra-class heterogeneity resorts to enhancing the synthetic images before feeding them to the pre-trained full-precision model F. Our motives lie in the fact that the objects of interest the model is expected to learn are not always at the same scale or position in the images. Thus, it is natural to expect the synthetic images to contain these informative contents at different scales or positions. However, earlier methods only focus on optimizing a complete image during the whole process of data generation given prior labels from the inception loss. Thus, the synthetic images tend to have the target objects all at the scale of covering up the whole images, and these synthetic images with the same prior label become very similar, which thus fails to retain the intra-class heterogeneity.

Motivated by the above analysis, we propose to locate the target objects at different scales and positions of the synthetic images. Specifically, for each synthetic image, instead of directly feeding the whole image to the pre-trained full-precision model F for optimization, we choose to randomly crop a patch of the image with a probability of p. For each cropping, its scaling rate is sampled from a uniform distribution $U(\eta, 1)$ where η is a hyper-parameter controlling the minimum scaling rate of the cropped patch. Thus, the input of the pre-trained full-precision F after our local object reinforcement becomes:

$$\tilde{\boldsymbol{I}}_{LOR} = \begin{cases} resize\big(crop_\eta(\tilde{\boldsymbol{I}})\big) & \text{with probability of } p \\ \tilde{\boldsymbol{I}} & \text{with probability of } 1-p, \end{cases} \tag{5}$$

where $crop_\eta(\cdot)$ randomly crops a patch from its input with a scaling rate sampled from a uniform distribution of $U(\eta, 1)$ where η is a pre-defined parameter, and $resize(\cdot)$ resizes its input to the size of the original synthetic image $\tilde{\boldsymbol{I}}$. The $p = 50\%$ is observed to perform best.

To stress, traditional cropping in data augmentation is to discard irrelevant contents and remain the pleasing portions of the image to enhance the overall composition. In contrast, our image cropping is to synthesize fake images with target objects at different scales and positions. As shown in Fig. 2, if cropped, the gradient is only backward to update the local cropped patch. Given a prior label, the cropping is applied at different positions and scales, and thus the synthesized images are no longer similar, which further retains the intra-class heterogeneity.

3.3.2 Marginal Distance Constraint

With the enhanced synthetic images as inputs, we can extract their feature vectors from the pre-trained full-precision network F. To correctly classify all the classes simultaneously, we expect F to form class-related features with large intra-class discrimination. To well generalize the quantized model to the real-world test dataset, we also expect F to form heterogeneous intra-class features. To achieve this, we further devise the following marginal distance constraint as a supervisory signal to guide the feature learning:

$$\begin{aligned}\mathcal{L}_{\text{MDC}}(\tilde{\boldsymbol{I}}_{LOR}) = max\Big(\lambda_l - cos\big(\mathcal{V}_F(\tilde{\boldsymbol{I}}_{LOR}), \mathcal{C}(\tilde{\boldsymbol{I}}_{LOR})\big), 0\Big) \\ + max\Big(cos\big(\mathcal{V}_F(\tilde{\boldsymbol{I}}_{LOR}), \mathcal{C}(\tilde{\boldsymbol{I}}_{LOR})\big) - \lambda_u, 0\Big),\end{aligned} \tag{6}$$

where $cos(\cdot, \cdot)$ returns the cosine distance of its two inputs in comparison and $\mathcal{V}_F(\cdot)$ returns the feature vector extracted by the pre-trained full-precision F, and $\mathcal{C}(\tilde{\boldsymbol{I}}_{LOR})$ returns the class center of $\tilde{\boldsymbol{I}}_{LOR}$. Assuming the label of $\tilde{\boldsymbol{I}}_{LOR}$ is c and M_c is a collection of the previously generated synthetic images belonging to class c, we define the class center as the mean feature vector of all synthetic images in M_c:

$$C(\tilde{\boldsymbol{I}}_{LOR}) = \frac{1}{|M_c|}\sum_{i=1}^{|M_c|} \mathcal{V}_F(\tilde{\boldsymbol{I}}_i), \;\; \tilde{\boldsymbol{I}}_i \in M_c. \tag{7}$$

The λ_l and λ_u in Eq. (6) are two hyper-parameters to control the lower and upper bounds of the cosine distance between $\tilde{\boldsymbol{I}}_{LOR}$ and its class center. Detailedly, Eq. (6) requires the distance larger than a margin λ_l, but smaller than a margin λ_u. The upper bound λ_u encourages features extracted from fake images of the same class to be similar, which brings about correct classification. The low bound λ_l avoids learning features concentrated on a dense area and thus can effectively preserve the intra-class heterogeneity, which ensures the generalization ability when the quantized model is employed on real-world test data.

3.3.3 Soft Inception Loss

The inception loss of Eq. (4) is to inject prior label knowledge into the synthetic images. To fulfill this goal, the loss essentially drives the gradients to optimize the synthetic images until the output of the pre-trained network F exactly matches the one-hot label. However, image contents are often overlapped even though they are grouped into different classes. One-hot labels do not represent soft decision boundaries among different objects, and hence the synthetic images trained on them are prone to overfitting to a fixed object. These images tend to be "easy" and do not well acquire the complex scenes within the contents. Consequently, existing methods embedded with the inception loss fail to retain the intra-class heterogeneity as shown in Fig. 1.

Reflecting on this, we consider soft labels as a regularization that has the potential to tell a model more about the meaning of each synthetic image. To be specific, given the enhanced synthetic image $\tilde{\boldsymbol{I}}_{LOR}$ with its prior label $y = c$, we devise the following soft inception loss:

$$\mathcal{L}_{\text{SIL}}(\tilde{\boldsymbol{I}}_{LOR}) = mse\big(F(\tilde{\boldsymbol{I}}_{LOR})_c, U(\epsilon, 1)\big), \tag{8}$$

where ϵ is a pre-defined parameter to control the softness of the label vector. Recall that $F(\cdot)$ returns the output of the softmax layer as stated in Sec. 3.1.2. Herein, $F(\cdot)_c$ indicates the c-th element of $F(\cdot)$. The $mse(\cdot, \cdot)$ computes the mean squared error between its two inputs.

Our soft inception loss requires the prediction probability of each synthetic image to match a soft label randomly sampled from a uniform distribution of $U(\epsilon, 1)$ instead of the hard one-hot form. Consequently, the synthetic images no longer overfit to a fixed object labeled with $y = c$, and more complex scenes are excavated, which further benefits the desired property of intra-class heterogeneity.

3.4. Training Process

Our learning of a quantized network consists of two parts including a data generation for fake images and fine-tuning of the quantized network upon the fake images.

3.4.1 Data Generation

We start with random input data $\tilde{\boldsymbol{I}}$ drawn from a standard Gaussian distribution. Our data generation aims to optimize $\tilde{\boldsymbol{I}}$, such that the distribution of the fake data can match that of real data in particular with the intra-class heterogeneity. To this end, as shown in Fig. 2, we first apply our local object reinforcement detailed in Sec. 3.3.1 to derive $\tilde{\boldsymbol{I}}_{LOR}$. Then, we feed $\tilde{\boldsymbol{I}}_{LOR}$ to the pre-trained full-precision network F to compute the BNS alignment loss of Eq. (3) and our proposed marginal distance constraint of Eq. (6). Also, we replace the traditional inception of Eq. (4) with our proposed soft inception loss of Eq. (8). As such, our final loss for data generation can be obtained as:

$$\mathcal{L}(\tilde{\boldsymbol{I}}_{LOR}) = \mathcal{L}_{\text{BNS}}(\tilde{\boldsymbol{I}}_{LOR}) + \mathcal{L}_{\text{MDC}}(\tilde{\boldsymbol{I}}_{LOR}) + \mathcal{L}_{\text{SIL}}(\tilde{\boldsymbol{I}}_{LOR}). \tag{9}$$

3.4.2 Network Fine-Tuning

With our synthetic fake images $\tilde{\boldsymbol{I}}$, we apply them to fine-tune the quantized network Q with the cross-entropy loss:

$$\mathcal{L}^{Q}_{\text{CE}} = ce\big(Q(\tilde{\boldsymbol{I}}), y\big). \tag{10}$$

Following [41], we also transfer the output of F to Q as:

$$\mathcal{L}^{Q}_{\text{KD}} = kl\big(Q(\tilde{\boldsymbol{I}}), F(\tilde{\boldsymbol{I}})\big), \tag{11}$$

where $kl(\cdot,\cdot)$ computes the Kullback-Leibler distance between its two inputs. Thus, the overall loss for fine-tuning the quantized network Q can be summarized as:

$$\mathcal{L}^{Q} = \mathcal{L}^{Q}_{\text{CE}} + \alpha \cdot \mathcal{L}^{Q}_{\text{KD}}, \tag{12}$$

where α balances the importance of $\mathcal{L}^{Q}_{\text{CE}}$ and $\mathcal{L}^{Q}_{\text{KD}}$.

In Tab. 1, our IntraQ results in an average intra-class cosine distance of 0.42, very close to 0.44 of the real data. Besides, the visualization results show that the distribution of our synthetic images (Fig. 1f) also approximates that of real data. Moreover, IntraQ obtains 66.47% in the top-1 accuracy, over 3.0% increases compared with ZeroQ and DSG integrated with the inception loss, as illustrated in Tab. 1.

4. Experiments

4.1. Implementation Details

We report top-1 accuracy on the validation set of CIFAR-10/100 [21] and ImageNet [36]. The quantized networks include ResNet-20 [10] for CIFAR-10/100, ResNet-18 [10], MobileNetV1 [15] and MobileNetV2 [37] for ImageNet. All experiments are implemented with Pytorch [33].

For data generation, Adam [19] is adopted with a momentum of 0.9 and the initial learning rate of 0.5. We update the synthetic images for 1,000 iterations and decay the learning rate by 0.1 each time the data generation loss of Eq. (9) stops decreasing for 50 iterations. The batch size is set as 256. There are four hyper-parameters in our data generation, including η in Eq. (5), λ_l and λ_u in Eq. (6), and ϵ in Eq. (8). They are respectively set to 0.5, 0.05, 0.8, and 0.9 on CIFAR-10; 0.5, 0.02, 1.0, and 0.6 on CIFAR-100; 0.5, 0.3 0.8, and 0.9 on ImageNet. As for ZeroQ+IL and DSG+IL, we implement the experiments based on their open-source code and use the same configurations as ours.

For all datasets, we generate 5,120 synthetic images to fine-tune the quantized model using SGD with Nesterov [32]. We set the weight decay as 10^{-4} and a total of 150 fine-tuning epochs are given. The batch size for fine-tuning is 256 for CIFAR-10/100 and 16 for ImageNet. Besides, CIFAR-10/100 is in configuration with an initial learning rate of 10^{-4} while it is 10^{-6} for ImageNet. Both learning rates are decayed by 0.1 every 100 fine-tuning epochs. The hyper-parameter in our network fine-tuning is α in Eq. (12) which is always set to 20.

Bit-width	Method	Generator	Acc. (%)
	full-precision	-	94.03
W4A4	Real data	-	91.52
	GDFQ	✓	90.25
	ZeroQ	✗	84.68
	DSG	✗	88.74
	ZeroQ+IL	✗	89.66
	DSG+IL	✗	88.93
	GZNQ	✗	91.30
	IntraQ (Ours)	✗	**91.49**
W3A3	Real data	-	87.94
	GDFQ	✓	71.10
	ZeroQ	✗	29.32
	DSG	✗	32.90
	ZeroQ+IL	✗	69.53
	DSG+IL	✗	48.99
	IntraQ (Ours)	✗	**77.07**

(a) CIFAR-10

Bit-width	Method	Generator	Acc. (%)
	full-precision	-	70.33
W4A4	Real data	-	66.80
	GDFQ	✓	63.58
	DSG	✗	62.36
	ZeroQ	✗	58.42
	DSG+IL	✗	62.62
	ZeroQ+IL	✗	63.97
	GZNQ	✗	64.37
	IntraQ (Ours)	✗	**64.98**
W3A3	Real data	-	56.26
	GDFQ	✓	43.87
	DSG	✗	25.48
	ZeroQ	✗	15.38
	DSG+IL	✗	43.42
	ZeroQ+IL	✗	26.35
	IntraQ (Ours)	✗	**48.25**

(b) CIFAR-100

Table 2. Results of ResNet-20 on CIFAR-10/100. WBAB indicates the weights and activations are quantized to B-bit.

4.2. Performance Comparison

4.2.1 CIFAR-10/100

We analyze the performance on CIFAR-10/100, comparing it against the popular ZSQ methods including GDFQ [41], ZeroQ [3], DSG [46] and GZNQ [11]. To demonstrate the efficacy, we quantize all layers of ResNet-20 to the ultra-low precisions of 4-bit and 3-bit since CIFAR-10/100 are relative simple datasets and high performance can be easily reached if larger quantization bits are given.

Bit-width	Method	Generator	Acc. (%)
	full-precision	-	71.47
W5A5	Real data	-	70.31
	GDFQ	✓	66.82
	DSG	✗	69.53
	ZeroQ	✗	69.65
	DSG+IL	✗	69.53
	ZeroQ+IL	✗	69.72
	IntraQ (Ours)	✗	**69.94**
W4A4	Real data	-	67.89
	GDFQ	✓	60.60
	DSG+G	✓	61.58
	ZeroQ	✗	60.68
	DSG	✗	60.12
	ZeroQ+IL	✗	63.38
	DSG+IL	✗	63.11
	GZNQ	✗	64.50
	IntraQ (Ours)	✗	**66.47**

Table 3. Results of ResNet-18 on ImageNet. WBAB indicates the weights and activations are quantized to B-bit.

From Tab. 2, we can see that our IntraQ consistently outperforms the compared methods on both CIFAR-10 and CIFAR-100. Specifically, compared to the advanced generator-based GDFQ, our IntraQ increases the top-1 accuracy of 3-bit quantized models by 5.97% on CIFAR-10 and 4.38% on CIFAR-100. Similar results can be observed in 4-bit quantization as well. In particular, compared with GZNQ [11] that requires 50,000 synthetic images to obtain accuracies of 91.30% and 64.37% on CIFAR-10 and CIFAR-100, the proposed IntraQ reaches the higher performance of 91.49% and 64.98% using only 5,120 synthetic images, well demonstrating the superiority of exploiting the intra-class heterogeneity in the synthetic fake images.

4.2.2 ImageNet

We further compare with the competitors on the large-scale ImageNet. The quantized networks include ResNet-18 and MobileNetV1/V2. Similar to CIFAR-10/100, we quantize all layers of the networks. Differently, we display the results of 5-bit and 4-bit due to the large scale of ImageNet.

ResNet-18. Tab. 3 shows the experimental results of ResNet-18. In the case of 5-bit, our IntraQ slightly outperforms the existing method ZeroQ with inception loss (69.94% *vs*. 69.72%). When it comes to 4-bit, a noticeable increase is observed from the proposed method. Detailedly, GZNQ obtains a limited accuracy of 64.50% using a total of 100,000 synthetic images. On the contrary, our IntraQ retains a high performance of 66.47% using only 5,120 synthetic images for fine-tuning the quantized ResNet-18, leading to 1.97% accuracy increases.

Bit-width	Method	Generator	Acc. (%)
	full-precision	-	73.39
W5A5	Real data	-	69.87
	GDFQ	✓	59.76
	ZeroQ	✗	61.95
	DSG	✗	64.18
	ZeroQ+IL	✗	67.11
	DSG+IL	✗	66.61
	IntraQ (Ours)	✗	**68.17**
W4A4	Real data	-	59.66
	GDFQ	✓	28.64
	ZeroQ	✗	20.96
	DSG	✗	21.14
	ZeroQ+IL	✗	25.43
	DSG+IL	✗	42.19
	IntraQ (Ours)	✗	**51.36**

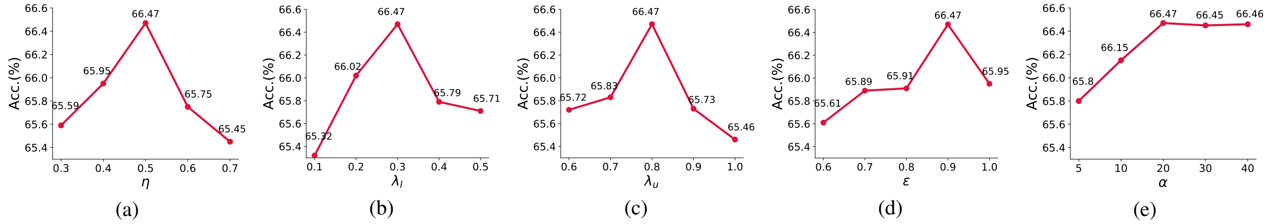
(a) MobileNetV1

Bit-width	Method	Generator	Acc. (%)
	full-precision	-	73.03
W5A5	Real data	-	72.01
	GDFQ	✓	68.14
	ZeroQ	✗	70.88
	DSG	✗	70.85
	ZeroQ+IL	✗	70.95
	DSG+IL	✗	70.87
	IntraQ (Ours)	✗	**71.28**
W4A4	Real data	-	67.90
	GDFQ	✓	51.30
	DSG+G	✓	54.66
	GZNQ	✗	53.53
	ZeroQ	✗	59.39
	DSG	✗	59.04
	ZeroQ+IL	✗	60.15
	DSG+IL	✗	60.45
	IntraQ (Ours)	✗	**65.10**

(b) MobileNetV2

Table 4. Results of MobileNetV1/V2 on ImageNet. WBAB indicates the weights and activations are quantized to B-bit.

MobileNetV1/V2. In Tab. 4, compared with ZeroQ+IL in 5-bit and DSG+IL in 4-bit, our IntraQ still maintains the best performance in quantizing the light-weight MobileNetV1/V2. The supreme performance is in particular obvious in lower 4-bit. For instance, our IntraQ obtains 9.17% accuracy improvements compared with DZSGQ+IL when all layers of MobileNetV1 are represented in a 4-bit form. These results again demonstrate the efficacy of our synthetic images for ZSQ and also verify the correctness of our motive to excavate the intra-class heterogeneity.



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

Figure 3. Influence of the hyper-parameters to the top-1 accuracy of 4-bit ResNet-18 on ImageNet.

LOR	MDC	SIL	Acc. (%)
ZeroQ+IL			63.38
✓			66.14
	✓		63.77
		✓	63.60
	✓	✓	64.05
✓	✓		66.32
✓		✓	66.30
✓	✓	✓	**66.47**

Table 5. Ablations on different components of our IntraQ. "LOR" indicates the local object reinforcement, "MDC" indicates the marginal distance constraint, and "SIL" indicates the soft inception loss. We report the top-1 accuracy of 4-bit ResNet on ImageNet.

4.3. Ablation Study

In this section, we conduct ablation studies of the hyper-parameters and different components of our IntraQ. All experiments are conducted by quantizing all layers of ResNet-18 to 4-bit on ImageNet. The top-1 accuracy is reported.

Hyper-parameters. The hyper-parameters include η in Eq. (5), λ_l and λ_u in Eq. (6), ϵ in Eq. (8), and α in Eq. (12). As shown in Fig. 3, the optimal results are η = 0.5, λ_l = 0.3, λ_u = 0.8, ϵ = 0.9, and η = 20. To avoid a cumbersome search, these results are used for all experiments on ImageNet. Though not optimal for all networks, they already show the best compared with existing methods. Similar experiments can be conducted to find out the optimal values of these parameters on other datasets, as listed in Sec. 4.1.

Components. We further study the effectiveness of our proposed local object reinforcement in Sec. 3.3.1, marginal distance constraint in Sec. 3.3.2, and soft inception loss in Sec. 3.3.3. Tab. 5 shows the experimental results. Note that ZeroQ+IL can serve as a baseline since it uses BNS alignment loss in Eq. (3) and inception loss in Eq. (4). As can be seen, when the three strategies are individually added to synthesize fake images, the accuracy increases compared with the baseline of ZeroQ+IL. Among them, the local object reinforcement significantly boosts the baseline from 63.38% to 66.14%. This inspires us of the importance of synthesizing images with objects in different scales and positions in order to retain the intra-class heterogeneity. Furthermore, the performance continues to increase if two of them are used together. When all of the three strategies are applied, the best performance of 66.47% can be obtained.

5. Limitations

Though the proposed IntraQ improves the accuracy of existing ZSQ methods by a large margin, its performance still degrades a lot if compared with the results of real data. Thus, how to further improve the quality of fake data remains to be investigated in our future work. Due to our limited hardware resources, we are unable to perform our IntraQ on other computer vision tasks (*e.g.*, detection). It is unclear whether the intra-class heterogeneity can still be observed, thus the applicability of our InterQ on other tasks remains an open issue. More efforts are required to address this problem in our near future.

6. Conclusion

In this paper, we investigate optimizing synthetic images for zero-shot quantization (ZSQ). We discover a non-ignorable phenomenon of intra-class heterogeneity in real data. To retain this property in the synthetic images for better performance, we propose a novel ZSQ method, called IntraQ. To that effect, our innovations are three folds including a local object reinforcement, a marginal distance constraint, and a soft inception loss. The local object reinforcement locates the target objects at different scales and positions of the synthetic images to avoid producing similar images. The marginal distance constraint is applied to prevent image features from being concentrated together. The soft inception loss considers a soft label as prior knowledge to excavate more complex scenes within the synthetic images. With our innovations, the synthetic images are demonstrated to be heterogeneous within each class and the quantized models fine-tuned on these images are experimentally shown to be superior in performance.

Acknowledgement. This work is supported by the National Science Fund for Distinguished Young Scholars (No.62025603), the National Natural Science Foundation of China (No.U1705262, No.62072386, No.62072387, No.62072389, No.62002305, No.61772443, No.61802324 and No.61702136), Guangdong Basic and Applied Basic Research Foundation (No.2019B1515120049) and the Fundamental Research Funds for the central universities (No.20720200077, No.20720200090 and No.20720200091).

References

[1] Ron Banner, Yury Nahshan, Daniel Soudry, et al. Post training 4-bit quantization of convolutional networks for rapid-deployment. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, pages 7950–7958, 2019. 2, 3

[2] Adrian Bulat, Brais Martinez, and Georgios Tzimiropoulos. High-capacity expert binary networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020. 1

[3] Yaohui Cai, Zhewei Yao, Zhen Dong, Amir Gholami, Michael W Mahoney, and Kurt Keutzer. Zeroq: A novel zero shot quantization framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13169–13178, 2020. 1, 2, 3, 6

[4] Yoojin Choi, Jihwan Choi, Mostafa El-Khamy, and Jungwon Lee. Data-free network quantization with adversarial knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 710–711, 2020. 2, 3

[5] Carl Doersch. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*, 2016. 3

[6] Steven K. Esser, Jeffrey L. McKinstry, Deepika Bablani, Rathinakumar Appuswamy, and Dharmendra S. Modha. Learned step size quantization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020. 2

[7] Jun Fang, Ali Shafiee, Hamzah Abdel-Aziz, David Thorsley, Georgios Georgiadis, and Joseph H Hassoun. Post-training piecewise linear quantization for deep neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 69–86, 2020. 2

[8] Ruihao Gong, Xianglong Liu, Shenghu Jiang, Tianxiang Li, Peng Hu, Jiazhen Lin, Fengwei Yu, and Junjie Yan. Differentiable soft quantization: Bridging full-precision and low-bit neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4852–4861, 2019. 1, 2

[9] Matan Haroush, Itay Hubara, Elad Hoffer, and Daniel Soudry. The knowledge within: Methods for data-free model compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8494–8502, 2020. 2, 3

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 6

[11] Xiangyu He, Jiahao Lu, Weixiang Xu, Qinghao Hu, Peisong Wang, and Jian Cheng. Generative zero-shot network quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 3000–3011, 2021. 1, 3, 6, 7

[12] Yang He, Ping Liu, Ziwei Wang, Zhilan Hu, and Yi Yang. Filter pruning via geometric median for deep convolutional neural networks acceleration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4340–4349, 2019. 1

[13] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 1

[14] Maxwell Horton, Yanzi Jin, Ali Farhadi, and Mohammad Rastegari. Layer-wise data-free cnn compression. *arXiv preprint arXiv:2011.09058*, 2020. 3

[15] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 6

[16] Qing Jin, Linjie Yang, and Zhenyu Liao. Adabits: Neural network quantization with adaptive bit-widths. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2146–2156, 2020. 2

[17] Sangil Jung, Changyong Son, Seohyung Lee, Jinwoo Son, Jae-Joon Han, Youngjun Kwak, Sung Ju Hwang, and Changkyu Choi. Learning to quantize deep networks by optimizing quantization intervals with task loss. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4350–4359, 2019. 2

[18] Dohyung Kim, Junghyup Lee, and Bumsub Ham. Distance-aware quantization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5271–5280, 2021. 1

[19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2014. 6

[20] Raghuraman Krishnamoorthi. Quantizing deep convolutional networks for efficient inference: A whitepaper. *arXiv preprint arXiv:1806.08342*, 2018. 1

[21] Alex Krizhevsky. Learning multiple layers of features from tiny images. *University of Toronto*, 2009. 6

[22] Junghyup Lee, Dohyung Kim, and Bumsub Ham. Network quantization with element-wise gradient scaling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6448–6457, 2021. 2

[23] Yuhang Li, Xin Dong, and Wei Wang. Additive powers-of-two quantization: An efficient non-uniform discretization for neural networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020. 1, 2

[24] Yuhang Li, Ruihao Gong, Xu Tan, Yang Yang, Peng Hu, Qi Zhang, Fengwei Yu, Wei Wang, and Shi Gu. Brecq: Pushing the limit of post-training quantization by block reconstruction. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 1, 2

[25] Mingbao Lin, Rongrong Ji, Yan Wang, Yichen Zhang, Baochang Zhang, Yonghong Tian, and Ling Shao. Hrank: Filter pruning using high-rank feature map. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1529–1538, 2020. 1

[26] Mingbao Lin, Rongrong Ji, Zihan Xu, Baochang Zhang, Yan Wang, Yongjian Wu, Feiyue Huang, and Chia-Wen Lin. Rotated binary neural network. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, pages 7474–7485, 2020. 2

[27] Xingchao Liu, Mao Ye, Dengyong Zhou, and Qiang Liu. Post-training quantization with multiple points: Mixed precision without mixed precision. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 8697–8705, 2021. 2

[28] Yuang Liu, Wei Zhang, and Jun Wang. Zero-shot adversarial quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1512–1521, 2021. 2, 3

[29] Brais Martinez, Jing Yang, Adrian Bulat, and Georgios Tzimiropoulos. Training binary neural networks with real-to-binary convolutions. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020. 2

[30] Markus Nagel, Rana Ali Amjad, Mart Van Baalen, Christos Louizos, and Tijmen Blankevoort. Up or down? adaptive rounding for post-training quantization. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 7197–7206, 2020. 1, 2

[31] Markus Nagel, Mart Van Baalen, Tijmen Blankevoort, and Max Welling. Data-free quantization through weight equalization and bias correction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1325–1334, 2019. 1, 3

[32] Yurii Evgen'evich Nesterov. A method of solving a convex programming problem with convergence rate o(k^2). In *Proceedings of the Russian Academy of Sciences (RAS)*, pages 543–547, 1983. 6

[33] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, pages 8026–8037, 2019. 6

[34] Haotong Qin, Ruihao Gong, Xianglong Liu, Mingzhu Shen, Ziran Wei, Fengwei Yu, and Jingkuan Song. Forward and backward information retention for accurate binary neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2250–2259, 2020. 2

[35] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 525–542, 2016. 2

[36] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, 115:211–252, 2015. 1, 6

[37] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4510–4520, 2018. 6

[38] Jianghao Shen, Yue Wang, Pengfei Xu, Yonggan Fu, Zhangyang Wang, and Yingyan Lin. Fractional skipping: Towards finer-grained dynamic cnn inference. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 5700–5708, 2020. 2

[39] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research (JMLR)*, 9:2579–2605, 2008. 2

[40] Peisong Wang, Qiang Chen, Xiangyu He, and Jian Cheng. Towards accurate post-training network quantization via bit-split and stitching. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 9847–9856, 2020. 1, 2

[41] Shoukai Xu, Haokun Li, Bohan Zhuang, Jing Liu, Jiezhang Cao, Chuangrun Liang, and Mingkui Tan. Generative low-bitwidth data free quantization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 1–17, 2020. 1, 2, 3, 6

[42] Jiwei Yang, Xu Shen, Jun Xing, Xinmei Tian, Houqiang Li, Bing Deng, Jianqiang Huang, and Xian-sheng Hua. Quantization networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7308–7316, 2019. 2

[43] Linjie Yang and Qing Jin. Fracbits: Mixed precision quantization via fractional bit-widths. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 10612–10620, 2021. 1

[44] Hongxu Yin, Pavlo Molchanov, Jose M Alvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8715–8724, 2020. 2

[45] Haichao Yu, Haoxiang Li, Humphrey Shi, Thomas S Huang, and Gang Hua. Any-precision deep neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 10763–10771, 2021. 2

[46] Xiangguo Zhang, Haotong Qin, Yifu Ding, Ruihao Gong, Qinghua Yan, Renshuai Tao, Yuhang Li, Fengwei Yu, and Xianglong Liu. Diversifying sample generation for accurate data-free quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15658–15667, 2021. 1, 2, 3, 6

[47] Bohan Zhuang, Lingqiao Liu, Mingkui Tan, Chunhua Shen, and Ian Reid. Training quantized neural networks with a full-precision auxiliary module. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1488–1497, 2020. 1, 2