Generative Zero-shot Network Quantization

Xiangyu He[*]   Jiahao Lu[*]   Weixiang Xu   Qinghao Hu   Peisong Wang   Jian Cheng

Institute of Automation Chinese Academy of Sciences, Beijing, China

{xiangyu.he, peisong.wang, jcheng}@nlpr.ia.ac.cn

Abstract

Convolutional neural networks are able to learn realistic image priors from numerous training samples in low-level image generation and restoration [66]. We show that, for high-level image recognition tasks, we can further reconstruct "realistic" images of each category by leveraging intrinsic Batch Normalization (BN) statistics without any training data. Inspired by the popular VAE/GAN methods, we regard the zero-shot optimization process of synthetic images as generative modeling to match the distribution of BN statistics. The generated images serve as a calibration set for the following zero-shot network quantizations. Our method meets the needs for quantizing models based on sensitive information, e.g., due to privacy concerns, no data is available. Extensive experiments on benchmark datasets show that, with the help of generated data, our approach consistently outperforms existing data-free quantization methods.

1. Introduction

Deep convolutional neural networks have achieved great success in several computer vision tasks [26, 61], however, we still understand little why it performs well. There are plenty of pioneering works aim at peeking inside these networks. Feature visualization [57, 51, 56] is a group of mainstream methods that enhance an input image from noise to elicit a particular interpretation. The generated images illustrate how neural networks build up their understanding of images [57], as a byproduct, opens a path towards data-free model compression [70].

Network quantization is an efficient and effective way to compress deep neural networks with a small memory footprint and low latency. It is a common practice to introduce a calibration set to quantize activations. To recover the degraded accuracy, training-aware quantization even requires re-training on the labeled dataset. However, in real-world applications, the original (labeled) data is commonly

---

*Equal contribution.

not available due to privacy and security concerns. In this case, zero-shot/data-free quantization becomes indispensable. The following question is *how to sample data x from the finite dataset $\mathcal{X}$ in the absence of original training data.*

It is intuitive to introduce noise $u \sim \mathcal{N}(0,1)$ as the input data to estimate the distributions of intermediate layers. Unfortunately, since the Single-Gaussian assumption can be too simple, the results for low-bit activation quantization are far from satisfactory [69]. Due to the zero-shot setting, it is also hard to apply the well-developed GANs to image synthesis without learning the image prior to the original dataset. Very recently, a large body of works suggests that the running mean μ and variance σ^2 in the Batch Normalization layer have captured the prior distribution of the data [11, 25, 15, 69]. The square loss on μ and σ^2 (details in Equation (11) and (10)) coupled with cross-entropy loss achieves the empirical success in zero-shot network quantization. Though the performance has been further improved over early works [7, 47, 57, 56] such as DeepDream [51], it remains unclear why these learning targets should lead to meaningful data after training. Therefore, instead of directly presenting several loss functions, we hope to better describe the training process via generative modeling, which might provide another insight into the zero-shot network quantization.

In this work, we consider the generative modeling that deals with the distributions of mean and variance in Batch Normalization layers [35]. That is, for some random data augmentations determined by random variable z, the mean μ and variance σ^2 generated by synthesized images I should look like their counterparts extracted from real images, with high probability. Recently developed generative models like GAN commonly captures distribution $p(I)$ instead of knowing one image, but we regard synthesized images I as model parameters optimized in zero-shot learning. The input transformations introduce the randomness to allow the sampling on μ and σ^2. Besides, our method presents a potential interpretation for the popular Batch Normalization matching loss [11, 25, 15, 69, 70]. Due to the insufficient sampling in each iteration, we further propose a prior regularization term to narrow the parameter space of I.

Since the accuracy drop of low-bit activation quantization heavily relies on the image quality of the calibration set, we conduct extensive experiments on the zero-shot network quantization task. The 4-bit networks including weights and activations show consistent improvements over baseline methods on both CIFAR and ImageNet. Hopefully, this work may not be restricted to image synthesis, but shed some light on the interpretability of CNN through generating interpretable input images that agree with deep networks' internal knowledge.

2. Related Work

Model quantization has been one of the workhorses in industrial applications which enjoys low memory footprint and inference latency. Due to its great practical value, low-bit quantization becomes popular in recent literature.

Training-aware quantization. Previous works [52, 24, 34, 75] mainly focus on recovering the accuracy of quantized model via backward propagations, *i.e.*, label-based fine-tuning. Since the training process is similar to its floating point counterpart, the following works further prove that low-bit networks can still achieve comparable performances with full-precision networks by training from scratch [18, 74, 45, 36]. Ultra low-bit networks such as binary [23, 17, 33, 60] and ternary [2, 13, 41, 76], benefiting from bitwise operations to replace the computing-intensive MACs, is another group of methods. Leading schemes have reached less than five points accuracy drop on ImageNet [42, 28, 48, 38]. While training-aware methods achieve good performance, they suffer from the inevitable re-training with enormous labeled data.

Label-free quantization. Since a sufficiently large open training dataset is inaccessible for many real-world applications, such as medical diagnosis [19], drug discovery, and toxicology [10], it is imperative to avoid retraining or require no training data. Label-free methods take a step forward by only relying on limited unlabeled data [27, 53, 4, 73, 3, 16]. Most works share the idea of minimizing the quantization error or matching the distribution of full precision weights via quantized parameters. [4, 22, 67, 53] observe that the changes of layer output play a core role in accuracy drop. By introducing the "bias correction" technique (minimize the differences between $\mathbb{E}(\boldsymbol{W}x)$ and $\mathbb{E}(\widehat{\boldsymbol{W}}x)$), the performance of quantized model can be further improved.

Zero-shot quantization. Recent works show that deep neural networks pre-trained on classification tasks can learn the prior knowledge of the underlying data distribution [66]. The statistics of intermediate features, *i.e.,* "metadata", are assumed to be provided and help to discriminate samples from the same image domain as the original dataset [7, 44]. To circumvent the need for extra "metadata", [55] treats the weights of the last fully-connected layer as the class
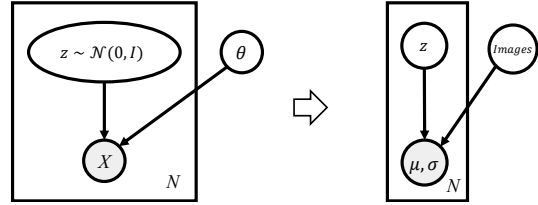

Figure 1: The standard VAE represented as a graphical model (left subfigure), *i.e.*, sampling from z N times to generate something similar to X with fixed parameters θ [20]. In this work, we regard the synthetic sampes/images I as model parameters θ to be optimized during the training process. We have a family of deterministic functions $f_I(z)$ parameterized by I. Since z is random, $f_I(z)$ is a random variable [1]. We hope to optimize I such that $\forall z_i$ from $p(z)$, $f_I(z_i)$ can cause the pre-trained network to generate μ, σ^2 and, with high probability, these random variables will be like the Batch-Normalization statistics [35] generated by real images.

templates then exploits the class similarities learned by the teacher network via Dirichlet sampling. Another group of methods focus on the stored running statistics of the Batch Normalization layer [35]. [70, 25] produce synthetic images without generators by directly optimizing the input images through backward propagations. Given any input noises, [12, 49, 71, 15, 37, 11] introduces a generator network $\boldsymbol{g}_\theta$ that yields synthetic images to perform Knowledge Distillation [29] between teacher and student networks.

Since we have no access to the original training dataset in zero-shot learning, it is hard to find the optimal generator $\boldsymbol{g}_\theta$. Generator-based methods have to optimize $\boldsymbol{g}_\theta$ indirectly via KL divergence between categorical probability distributions instead of $\max \mathbb{E}_{x \sim \hat{p}}[\ln p(x)]$ in VAE/GANs. In light of this, we formulate the optimization of synthetic images I as generative modeling that maximizes $\mathbb{E}[\ln p(\mu, \sigma^2; I)]$.

3. Approach

3.1. Preliminary

A generative modeling whose purpose is to map random variables to samples and generates samples distributed according to $\hat{p}(x)$, defined over datapoints $\mathcal{X}$. We are interested in estimating $\hat{p}$ using maximum likelihood estimation to find best $p(x)$ that approximates $\hat{p}$ measured by Kullback-Leibler divergence,

$$\mathcal{L}(x; \theta) = \text{KL}(\hat{p}(x) || p(x; \theta)) = \mathbb{E}_{\hat{p}(x)}[\ln p(x; \theta)]. \quad (1)$$

Here we use a parametric model for distribution p. We hope to optimize θ such that we can maximize the log likelihood.

---

[1]Given z_i, we have a deterministic function $f_I(z_i)$ which applies flipping/jitter/shift to the synthetic images I according to z_i. We can sample z_i from probability density function $p(z)$ N times to allow the backprop.

Ideally, $p(x; \theta)$ should be sufficiently expressive and flexible to describe the true distribution $\hat{p}$. To this end, we introduce a latent variable z,

$$p(x; \theta) = \int p(x|z; \theta)p(z)dz = \mathbb{E}_{p(z)}[p(x|z; \theta)], \quad (2)$$

so that the marginal distribution p computed by the product of diverse distributions (*i.e.,* joint distribution) can better approximate $\hat{p}$.

3.2. Generative Zero-shot Quantization

We now describe the optimization procedure of Generative Zero-shot Quantization (GZNQ) with batch normalization and draw the resemblance to the generative modeling.

The feed forward function of a deep neural network at l-th layer can be described as:

$$F^l = \mathbf{W}^l \phi(\mathbf{W}^{l-1}...\phi(\mathbf{W}^2\phi(\mathbf{W}^1(f_I(z))))) \quad (3)$$

$$\mu_{\text{batch}} = \frac{\sum_{i=1}^{N} F_i^l}{N} \quad \sigma_{\text{batch}}^2 = \frac{\sum_{i=1}^{N}(F_i^l - \mu_{\text{batch}})^2}{N} \quad (4)$$

where $\phi(\cdot)$ is an element-wise nonlinearity function and $\mathbf{W}^l$ is the weights of l-th layer (freezed during the optimization process). The mean μ_{batch} and variance σ_{batch}^2 are calculated per-channel over the mini-batches. Furthermore, we denote the input to networks as $f_I(z)$. That is, we have a vector of latent variables z in some high-dimensional space $\mathcal{Z}$ [2] but we can easily sample z_i according to $p(z)$. Then, we have a family of deterministic functions $f_I(z_i)$, parameterized by the synthetic samples/images I in pixel space $\mathcal{I}$. z_i determines the parameters such as the number of places by which the pixels of the image are shifted and whether to apply flipping/jitter/shift function f to I. Though $I, \mathbf{W}$ are fixed and the mapping $\mathcal{Z} \times \mathcal{I} \rightarrow \mu, \sigma^2$ is deterministic, if z is random, then μ_{batch} and σ_{batch}^2 are random variables in the space of μ and σ^2. We wish to find the optimal I^* such that, even with random flipping/jitter/shift, the computed $\mu_{\text{batch}}, \sigma_{\text{batch}}^2$ are still very similar to the Batch-Normalization (BN) statistics [35] generated by real images.

Recall Equation (1), minimizing the KL divergence is equivalent to maximizing the following log likelihood

$$I^* = \arg\max_I \ln \mathbb{E}_{p(z)}[p(\mu, \sigma^2|z; I)]. \quad (5)$$

To perform stochastic gradient descent, we need to compute the expectation. However, taking the expectation with respect to $p(z)$ in closed form is not possible in practice. Instead, we take Monte Carlo (MC) estimation by sampling

from $p(z)$

$$\mathbb{E}_{p(z)}[p(\mu, \sigma^2|z; I)] \approx \frac{1}{n}\sum_{i=1}^{n} p(\mu, \sigma^2|z_i; I). \quad (6)$$

Then, we may approximate the distribution of mean and variance of a mini-batch, *i.e.,* $p(\mu)$ and $p(\sigma^2)$.

Distribution matching For the mean variable, we have $\mu_{\text{batch}} = \frac{\sum_{i=1}^{N} F_i}{N}$ where F_i are features in the sampled mini-batch. We assume that samples of the random variable are i.i.d. then by central limit theorem (CLT) we obtain

$$\mu_{\text{batch}} \sim \mathcal{N}(\mu, \frac{\sigma^2}{N}) \quad (7)$$

for sufficiently large N, given $\mu = \mathbb{E}[F]$ and $\sigma^2 = \mathbb{E}[(F - \mu_{\text{batch}})^2]$. Similarly, we get

$$\sigma_{\text{batch}}^2 \sim \mathcal{N}(\sigma^2, \frac{\text{Var}[(F^l - \mu)^2]}{N}), \quad (8)$$

where N accounts for the batchsize and $\text{Var}[\cdot]$ is the finite variance, details in Appendix. Then, we further rewrite Equation (5) through (6-8) as follows:

$$\mathcal{L}_{DM} = \underbrace{\frac{1}{2}\frac{N}{\sigma^2}(\mu_{\text{batch}} - \mu)^2 + \frac{1}{2}\frac{N}{\text{Var}[(F^l - \mu)^2]}(\sigma_{\text{batch}}^2 - \sigma^2)^2}_{\text{term I}}$$
$$+ \frac{1}{2}\ln \text{Var}[(F^l - \mu)^2]. \quad (9)$$

Rethinking BN matching Note that the popular Batch-Normalization matching loss in recent works [11, 70, 69]

$$\min ||\vec{\mu}_{\text{batch}} - \vec{\mu}||_2^2 + ||\vec{\sigma}_{\text{batch}}^2 - \vec{\sigma}^2||_2^2,$$

which can be regarded as a simplified term I in Eq.(9), leaving out the correlation between μ and σ^2 (*i.e.*, the coefficients). Another group of recent methods [15, 25] present

$$\min \log \frac{\mu_{\text{batch}}}{\mu} - \frac{1}{2}(1 - \frac{\sigma^2 + (\sigma^2 - \sigma_{\text{batch}}^2)^2}{\sigma_{\text{batch}}^2}), \quad (10)$$

which actually minimizes the following object

$$\min \text{KL}(\mathcal{N}(\mu, \sigma^2) \,||\, p(F^l)), \quad F^l \sim \mathcal{N}(\mu_{\text{batch}}, \sigma_{\text{batch}}^2).$$

That is to approximate the distribution $p(F^l)$ defined over features F^l instead of Batch-Normalization statistics μ, σ^2 in Eq.(5). Since the parameter space of featuremaps are much larger than μ, σ^2, we adopt $\mathcal{L}_{DM}$ to facilitate the learning process.

Pseudo-label generator Unfortunately, Monte Carlo estimation in (6) can be inaccurate given a limited sampling [3], which may lead to a large gradient variance and poor synthesized results. Hence, it is common practice to introduce regularizations via prior knowledge of I, *e.g.,*
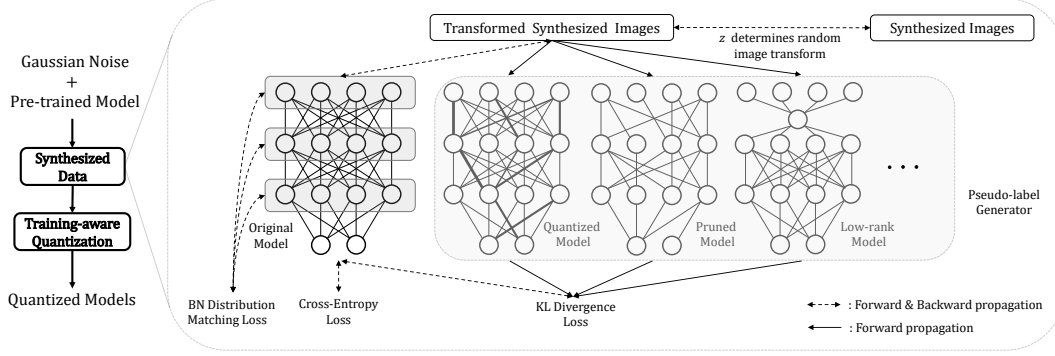
Figure 2: Generative Zero-shot Network Quantization (GZNQ) setup: GZNQ uses a generative model to produce the mean and variance in Batch-Normalization (BN) layer [35], meanwhile, optimizes the synthesized images to perform the following training-aware quantization. The pseudo-label generator consists of several data-free post-training compressed models (same architecture using different post-training compression schemes, *e.g.,* weights quantization + weights magnitude-based unstructured pruning + SVD decomposition), which serve as a multi-model ensemble or voting classifier.

$$\min \ \mathcal{L}_{CE}(\varphi_{\boldsymbol{\omega}}(I), y) \tag{11}$$

where $\varphi_{\boldsymbol{\omega}}(I)$ produces the categorical probability and y accounts for the ground-truth, which can be regarded as a prior regularization on I.

Recently, [31, 32] shows that compressed models are more vulnerable to challenging or complicated examples. Inspired by these findings, we wished to introduce a post-training quantized low-bit network as the pseudo-label generator to help produce "hard" samples. However, the selection of bitwidth can be tricky. High-bit networks yield nearly the same distribution as the full-precision counterpart, which results in noisy synthetic results (as pointed out in adversarial attack, high-frequency noise can easily fool the network). Low-bit alternatives fail on easy samples that damage the image diversity. To solve this, we turn to the model ensemble technique, shown in Figure 2, then reveal the similarity between (12) and multiple generators/discriminators training in GANs [30, 21].

An ensemble of different post-training compressed models generates a similar categorical distribution with the original network (illustrated in Figure 3) and it is more flexible to adjust the regularization strength than discrete bitwidth selection. Note that ensemble modeling still obtains a small KL distance when the accuracy is relatively low. Here, we get the prior regularization on I as

$$\mathcal{L}_{KL} = \mathrm{KL}\Big(\varphi_{\boldsymbol{\omega}}\left(f_I\left(z\right)\right) \| \frac{1}{M} \sum_{i=1}^{M} \varphi_{\hat{\boldsymbol{\omega}}^i}\left(f_I\left(z\right)\right)\Big)$$
$$= \sum_{j=1}^{N} \varphi_{\boldsymbol{\omega}_j}\left(f_I\left(z\right)\right) \log \frac{\varphi_{\boldsymbol{\omega}_j}\left(f_I\left(z\right)\right)}{\frac{1}{M} \sum_{i=1}^{M} \varphi_{\hat{\boldsymbol{\omega}}_j^i}\left(f_I\left(z\right)\right)} \tag{12}$$

where $\hat{\boldsymbol{\omega}}^i$ refers to the i-th compressed model. For notational simplicity, we shall in the remaining text denote

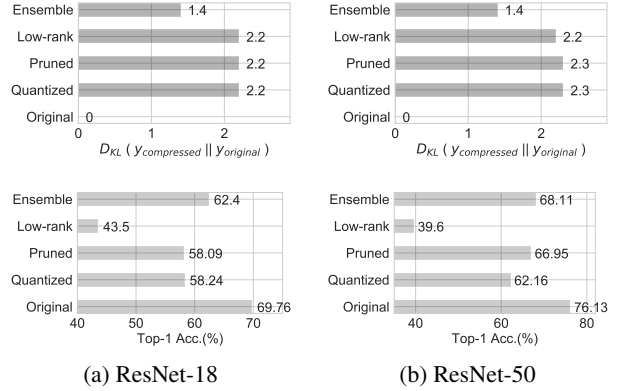(a) ResNet-18   (b) ResNet-50

Figure 3: An ensemble of compressed models generates a relatively reliable pseudo-label, which is more similar to the distribution of original network outputs than every single compressed model when the accuracy is comparable.

$\varphi_{\boldsymbol{\omega}_j}\left(f_I\left(z\right)\right)$ as $\varphi_{\boldsymbol{\omega}_j}$. By simply applying the AM-GM inequality to (12), we can easily prove that (12) serves as a "lower bound" for the objective with multiple compressed models

$$\sum_{j=1}^{N} \varphi_{\boldsymbol{\omega}_j} \log \frac{\varphi_{\boldsymbol{\omega}_j}}{\frac{1}{M} \sum_i \varphi_{\hat{\boldsymbol{\omega}}_j^i}} \leqslant \sum_{j=1}^{N} \varphi_{\boldsymbol{\omega}_j} \log \frac{\varphi_{\boldsymbol{\omega}_j}}{\left(\prod_i \varphi_{\hat{\boldsymbol{\omega}}_j^i}\right)^{\frac{1}{M}}}$$
$$= \frac{1}{M} \sum_{i=1}^{M} \mathrm{KL}\Big(\varphi_{\boldsymbol{\omega}}\left(f_I(z)\right) \| \varphi_{\hat{\boldsymbol{\omega}}^i}\left(f_I(z)\right)\Big). \tag{13}$$

[21] shows multiple discriminators can alleviate the mode collapse problem in GANs. Here, (13) encourages the synthesized images to be generally "hard" for all compressed models (*i.e.,* the large KL divergence corresponds to the disagreement between full-precision networks and compressed models).

channel gating matrix, $G \in \{0,1\}^{C \times N}$
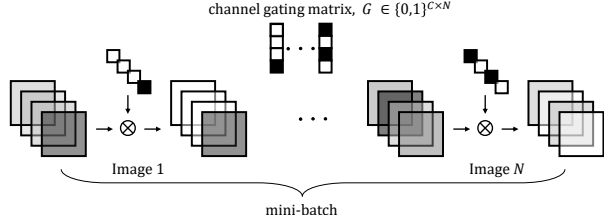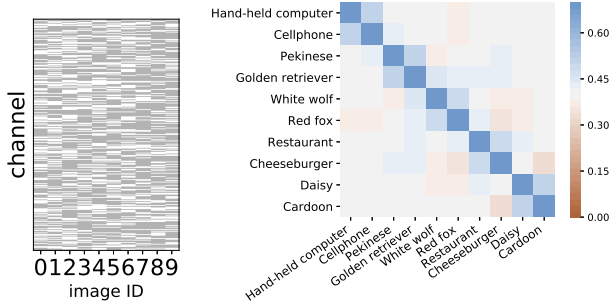
Image 1 ... Image N

mini-batch

Figure 4: Illustration of the channel gating module. Note that each sample has its own binary gating mask $G_i \in \{0,1\}^{1 \times c}$. In this case, $c_{channel} = 4$.

(a) channel gating   (b) correlation matrix

Figure 5: We visualize the learned channel gating mask of ten samples in a), generated by ImageNet ResNet-50. There are plenty of zero elements which illustrate the neural redundancy. We further show the correlation matrix of gating masks of ten samples in b). Images belonging to the same main category produce higher responses than irrelevant classes, *e.g.,* the gating mask of cardoon is more similar to daisy than cheeseburger.

Channel gating From the view of connectionism, "memory" is created by modifying the strength of the connections between neural units [64, 9], *i.e.,* weights matrix. In light of this, we wish to do the optimal surgery [40, 68] to enhance the "memory" when generating samples belonging to a specific category. More specifically, we use *per-sample* channel pruning to encourage learning more entangled representations, shown in Figure 4. Since channel pruning may severely damage the BN statistics, we only apply this setting to the last convolution layer. Fortunately, high-level neurons in deep CNNs are more semantically meaningful, which meets our needs.

We use the common Gumble-Softmax trick [46] to perform channel gatings

$$G_{i,j} = \begin{cases} 1, & \delta\left(\frac{\alpha_{i,j} + \log U - \log(1-U)}{\tau}\right) \geqslant 0 \\ 0, & \text{otherwise} \end{cases} \quad (14)$$

where δ is the sigmoid function, $U \sim \text{Uniform}(0,1)$ and τ is the temperature that controls the difference between the softmax and argmax function. Compared with channel

pruning [6] and DARTS [43], we introduce a 2D trainable matrix $\alpha \in \mathbb{R}^{C \times N}$ instead of a vector to better describe each sample/category. Figure 5b further illustrates the effect of channel gating that samples of the same main category yield similar masks after training.

4. Experiments

We perform experiments on the small-scale CIFAR-10/100 dataset (32×32 pixels) and the complex ImageNet dataset (224×224 pixels, 1k classes). The quantization and fine-tuning process are strictly data-free. We then report the Top-1 accuracy on the validation set.

4.1. Ablation study

In this section, we evaluate the effect of each component. All the ablation experiments are conducted on the ImageNet dataset with pre-trained standard ResNet-50 [26]. We use the popular Inception Score (IS) [62] to measure the sample quality despite its notable flaws [5]. As shown in Table 3, introducing prior regularization significantly contributes to higher inception scores and other components further improve IS. Since the activations become more semantically meaningful as layers go deeper, we apply (9) to the convolution layers in the last block of ResNets. Table 4 shows ensemble modeling leads to better performance than a single compressed model.

4.2. Generation details

z setting In this work, we assume z_i to be a three-dimensional random variable such as $z_{i,0} \sim \text{B}(0.5)$ and $z_{i,1}.z_{i,2} \sim \text{U}(-30,30)$, which determines whether to flip images and move images along any dimension by any number of pixels.

Pseudo-label generator setting We set the post-training quantized networks to 4-bit then adjust the percentage of rank to be preserved and pruning rate of each layer based on the KL divergence. That is, for every compressed model, the KL divergence between $y_{\text{compressed}}$ and y_{original} should be the same (or quite similar), as shown in Figure 3.

Training details As shown in Figure 2, GZNQ is a two-stage scheme. In this section, we detail the generative settings in the first stage. For CIFAR models, we first train full-precision networks from scratch (initial learning rate 0.1 with cosine scheduler; weight decay is $1e^{-4}$; all networks are trained for 300 epochs by SGD optimizer). Then, we utilize the proposed distribution matching loss, KL divergence loss, channel gating, and CE loss to optimize the synthesized images. More specifically, we use Adam optimizer (beta1 is set to 0.3 and beta2 is 0.9) with a learning rate of 0.4 and generate 32×32 images in a mini-batch of 128. The weight of BN matching loss is 0.03 and 0.05 for the KL divergence loss. We first half the image size to speed up training via 2×2 average downsampling. After 2k

Dataset	Pre-trained Model	Method	W bit	A bit	Quant Acc (%)	Acc Drop (%)	Fine-tuning
CIFAR-10	ResNet-20 (0.27M)	ZeroQ [11]	4	4	79.30	14.73	–
		Ours	4	4	89.06	4.07	–
		GDFQ [69]	4	4	90.25	3.78	✓
		Ours	4	4	**91.30**	1.83	✓
	ResNet-44 (0.66M)	Knowledge Within [25]	4	4	89.10	4.13	–
		Ours	4	4	91.46	2.92	–
		Knowledge Within [25]	$4^{\dagger}$	8	92.25	0.99	–
		Ours	4	8	**93.57**	0.83	–
	WRN40-2 (2.24M)	DFNQ [15]	4	8	94.22	0.55	✓
		Ours	4	4	**94.81**	0.37	✓
		DFNQ [15]	4	8	93.14	1.63	–
		Ours	4	4	94.06	1.09	–
CIFAR-100	ResNet-20 (0.28M)	ZeroQ [11]	4	4	45.20	25.13	–
		Ours	4	4	58.99	10.18	–
		GDFQ [69]	4	4	63.58	6.75	✓
		Ours	4	4	**64.37**	4.80	✓
	ResNet-18 (11.2M)	DFNQ [15]	4	8	75.15	2.17	✓
		Ours	4	5	**75.95**	3.16	✓
		DFNQ [15]	4	8	71.02	6.30	–
		Ours	4	5	71.15	7.96	–

Table 1: Results of zero-shot quantization methods on CIFAR-10/100. "W bit" means weights quantization bitwidth and "A bit" is quantization bits for activations. "Fine-tuning" refers to re-training on the generated images using knowledge distillation. † indicates first and last layers are in 8-bit. We directly cite the best results reported in the original zero-shot quantization papers (ZeroQ 4-bit activations from [69]).

BN	CE-Loss	Pseudo-label	BN+	Gating	Inception Score ↑
✓					7.4
✓	✓				43.7
✓	✓	✓			74.0
✓	✓	✓	✓		80.6
✓	✓	✓	✓	✓	**84.7**

Table 3: Impact of the proposed modules on GZNQ. "BN+" refers to Equation (9). Following GAN works [8, 72], we use Inception Score (IS) to evaluate the visual fidelity of generated images.

	Quant	Pruning	Low-rank	Ensemble Eq.(12)	Ensemble Eq.(13)
IS ↑	71.9±3.85	73.0±2.47	81.3±2.16	84.7±1.76	82.7±2.90

Table 4: Ablation study on the effect of model ensemble in the Pseudo-label generator. IS stands for Inception Score.

Calibration Dataset	Quantized Model Acc. (%)			
	WRN40-2 (CIFAR-10)	ResNet-20 (CIFAR-100)	ResNet-18 (ImageNet)	MobileNetv2 (ImageNet)
SVHN	50.11	5.43	3.00	1.50
CIFAR-100	92.94	**60.29**	14.50	7.56
CIFAR-10	**94.01**	57.29	14.10	6.46
ImageNet	80.72	38.75	**63.21**	**57.93**
Ours	94.06	58.99	57.24	54.03
FP32	93.13	95.18	69.76	71.78

Table 5: Impact of using different calibration datasets for 4-bit weights and 4-bit activation post-training quantization. Our synthesized dataset achieves comparable performance to the real images in most cases.

iterations, we use the full resolution images to optimize for another 2k iteration. We follow most settings of CIFAR-10/100 in ImageNet experiments. Since BN statistics and pseudo-labels are dataset-dependent, we adjust the weight of BN matching loss and KL divergence loss to 0.01 and 0.1 respectively. ImageNet pre-trained models are downloaded from torchvision model-zoo directly [58].

In our experiments, we do observe that networks trained only with random $256 \times N/N \times 256$ cropping and flipping contribute to high-quality images but the accuracy is relatively lower than the official model. This finding is consistent with the policy in differential privacy [1]. Since we focus on generative modeling and network quantization, more ablation studies on this part will be our future works.

4.3. Quantization details

Low-bit activation quantization typically requires a calibration set to constrain the dynamic range of intermediate layers to a finite set of fixed points. As shown in Table 5, it is crucial to collect images sampled from the same domain as the original dataset. To fully evaluate the effectiveness of GZNQ, we use synthetic images as the calibration set for sampling activations [39, 34], and BN statistics [59, 27].

Pre-trained Model	Method	W bit	A bit	Quant Top-1 (%)	Top-1 Drop (%)	Real-data	Fine-tuning
ResNet-50 (25.56M)	DoReFa [75]	4	4	71.4	5.5	✓	✓
	Ours	4	4	**72.7**	3.4	–	✓
	OMSE [16]	4	32	67.4	8.6	–	–
	Ours	4	6	68.1	8.1	–	–
	OCS [73]	6	6	74.8	1.3	✓	–
	ACIQ [3]	6	6	74.3	1.3	✓	–
	Ours	6	6	**75.5**	0.6	–	–
ResNet-18 (11.69M)	ZeroQ [11]	4	4	26.0	45.4	–	–
	Ours	4	4	57.2	12.5	–	–
	Knowledge Within [25]	4†	4	55.5	14.3	–	–
	Ours	4†	4	**58.9**	10.9	–	–
	GDFQ [69]	4	4	60.6	10.9	–	✓
	Ours	4	4	**64.5**	5.30	–	✓
	Integer-Only [36]	6	6	67.3	2.46	✓	✓
	DFQ [54]	6	6	66.3	3.46	–	–
	Ours	6	6	**69.0**	0.78	–	–
MobileNetv2 (3.51M)	Knowledge Within [25]	4‡	4	16.10	55.78	–	–
	Ours	4	4	**54.03**	17.72	–	–
	Integer-Only [36]	6	6	70.90	0.85	✓	✓
	Ours	6	6	**71.12**	0.63	–	✓
	DFQ [54]	8	8	71.19	0.38	–	–
	Knowledge Within [25]	8	8	71.32	0.56	–	–
	Ours	8	8	**71.38**	0.36	–	–

Table 2: Quantization results on ImageNet. "Real-data" means using original dataset as the calibration set to quantize activations or fine-tune weights. "Fine-tuning" refers to re-training with KD on the generated images (if no real data is required) or label-based fine-tuning. † indicates first and last layers are in 8-bit. ‡ means 8-bit 1×1 convolution layer. We directly cite the best results reported in the original papers (ZeroQ from [69]).

Method	Resolution	GAN	Inception Score ↑
BigGAN-deep [8]	256	✓	202.6
BigGAN [8]	256	✓	178.0
SAGAN [72]	128	✓	52.5
SNGAN [50]	128	✓	35.3
GZNQ	224	-	**84.7**±2.8
DeepInversion [70]	224	-	60.6
DeepDream [51]	224	-	6.2
ZeroQ [11]	224	-	2.8

Table 6: Inception Score (IS, higher is better) of various methods on ImageNet. SNGAN score reported in [63] and DeepDream score from [70]. The bottom four schemes are data-free and utlizing ImageNet pre-trained ResNet-50 to obtain synthesized images.

Besides, we use MSE quantizer [65, 27] for both weights and activations quantization, though, it is sensitive towards outliers. In all our experiments, floating-point per-kernel scaling factors for the weights and a per-tensor scale for the layer's activation values are considered. We keep a copy of full-precision weights to accumulate gradients then conduct quantization in the forward pass [34, 25]. Additionally, bias terms in convolution and fully-connected layers, gradients, and Batch-Normalization layers are kept in floating points. We argue that advanced calibration methods [53, 67, 69] or quantization schemes [16, 14, 36] coupled with GZNQ may

further improve the performance.

For our data-free training-aware quantization, *i.e.,* fine-tuning, we follow the setting in [69, 15] to utilize the vanilla Knowledge Distillation (KD) [29] between the original network and the compressed model. Since extra data augmentations can be the game-changer in fine-tuning, we use the common 4 pixels padding with 32×32 random cropping for CIFAR and the official PyTorch pre-processing for ImageNet, *without bells and whistles*. The initial learning rate for KD is 0.01 and trained for 300/100 epochs on CIFAR/ImageNet, decayed every $\frac{N}{3}$ iterations with a multiplier of 0.1. The batch size is 128 in all experiments. We also follow [70, 69] to fix batch normalization statistics during fine-tuning on ImageNet. All convolution and fully-connected layers are quantized to 4/6-bit, unless specified, including the first and last layer. The synthesized CIFAR dataset consists of 50k images and ImageNet has roughly 100 images per category.

4.4. Comparisons on benchmarks

We compare different zero-shot quantization methods [11, 25, 15, 69] on CIFAR-10/100 and show the results in Table 1. Our methods consistently outperform other state-of-the-art approaches in 4-bit settings. Furthermore, benefiting from the high quality generated images, the experimental results in Table 2 illustrate that, as the dataset gets larger, GZNQ still obtains notably improvements over base-
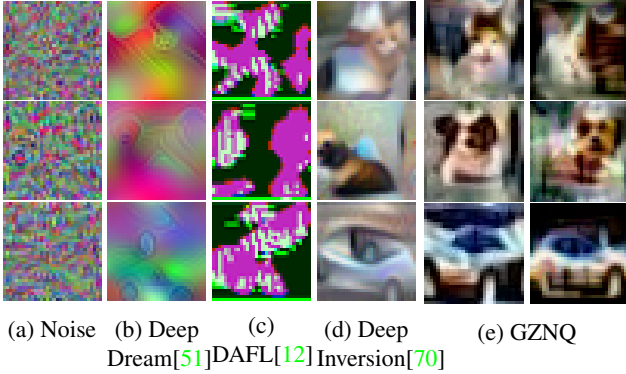
(a) Noise (b) Deep Dream[51] (c) DAFL[12] (d) Deep Inversion[70] (e) GZNQ

Figure 6: Synthetic samples generated by a CIFAR-10 pretrained ResNet-34 at a 32×32 resolution. We directly cite the best visualization results reported in [70].

line methods. Due to the different full-precision baseline accuracy reported in previous works, we also include the accuracy gap between floating-point networks and quantized networks in our comparisons. We directly cite the results in original papers to make a fair comparison, using the same architecture. In all experiments, we fine-tune the quantized models on the synthetic dataset generated by their corresponding full-precision networks.

We further compare our method with [11, 25, 12, 70] on the visual fidelity of images generated on CIFAR-10 and ImageNet. Figure 6-8 show that GZNQ is able to generate images with high fidelity and resolution. We observe that GZNQ images are more realistic than other competitors (appear like cartoon images). Following [70], we conduct quantitative analysis on the image quality via Inception S-core (IS) [62]. Our approach surpasses previous works, that is consistent with visualization results.

5. Conclusions

We present generative modeling to describe the image synthesis process in zero-shot quantization. Different from data-driven VAE/GANs that estimates $p(x)$ defined over real images $\mathcal{X}$, we focus on matching the distribution of mean and variance of Batch Normalization layers in the absence of original data. The proposed scheme further interprets the recent Batch Normalization matching loss and leads to high fidelity images. Through extensive experiments, we have shown that GZNQ performs well on the challenging zero-shot quantization task. The generated images also serve as an attempt to visualize what a deep convolutional neural network expects to see in real images.

References

[1] Martín Abadi, Andy Chu, Ian J. Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep

Figure 7: ImageNet samples generated by our GZNQ ResNet-50 model at 224×224 resolution : bear, daisy, balloon, pizza, stoplight, stingray, quill, volcano.
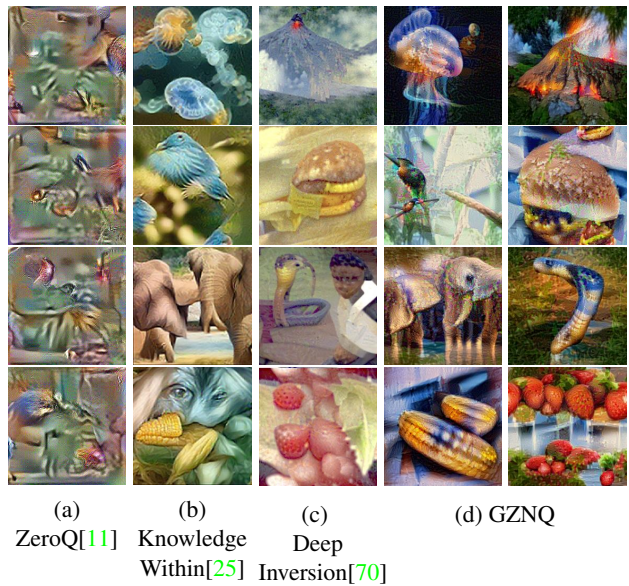

(a) ZeroQ[11] (b) Knowledge Within[25] (c) Deep Inversion[70] (d) GZNQ

Figure 8: Synthetic samples generated by a ImageNet pretrained ResNet-50 using different methods. We directly cite the best visualization results reported in the original papers.

learning with differential privacy. In Edgar R. Weippl, Stefan Katzenbeisser, Christopher Kruegel, Andrew C. Myers, and Shai Halevi, editors, *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, pages 308–318. ACM,

2016. 6

[2] Shai Abramson, David Saad, and Emanuel Marom. Training a network with ternary weights using the CHIR algorithm. *IEEE Trans. Neural Networks*, 4(6):997–1000, 1993. 2

[3] Ron Banner, Yury Nahshan, Elad Hoffer, and Daniel Soudry. ACIQ: analytical clipping for integer quantization of neural networks. *CoRR*, abs/1810.05723, 2018. 2, 7

[4] Ron Banner, Yury Nahshan, and Daniel Soudry. Post training 4-bit quantization of convolutional networks for rapid-deployment. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 7948–7956, 2019. 2

[5] Shane T. Barratt and Rishi Sharma. A note on the inception score. *CoRR*, abs/1801.01973, 2018. 5

[6] Babak Ehteshami Bejnordi, Tijmen Blankevoort, and Max Welling. Batch-shaping for learning conditional channel gated networks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. 5

[7] Kartikeya Bhardwaj, Naveen Suda, and Radu Marculescu. Dream distillation: A data-independent model compression framework. *CoRR*, abs/1905.07072, 2019. 1, 2

[8] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. 6, 7

[9] Cameron Buckner and James Garson. Connectionism. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, fall 2019 edition, 2019. 5

[10] Robert Burbidge, Matthew W. B. Trotter, Bernard F. Buxton, and Sean B. Holden. Drug design by machine learning: Support vector machines for pharmaceutical data analysis. *Computers & Chemistry*, 26(1):5–14, 2002. 2

[11] Yaohui Cai, Zhewei Yao, Zhen Dong, Amir Gholami, Michael W. Mahoney, and Kurt Keutzer. Zeroq: A novel zero shot quantization framework. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 13166–13175. IEEE, 2020. 1, 2, 3, 6, 7, 8

[12] Hanting Chen, Yunhe Wang, Chang Xu, Zhaohui Yang, Chuanjian Liu, Boxin Shi, Chunjing Xu, Chao Xu, and Qi Tian. Data-free learning of student networks. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 3513–3521. IEEE, 2019. 2, 8

[13] Tzi-Dar Chiueh and Rodney M. Goodman. Learning algorithms for neural networks with ternary weights. *Neural Networks*, 1(Supplement-1):166–167, 1988. 2

[14] Jungwook Choi, Zhuo Wang, Swagath Venkataramani, Pierce I-Jen Chuang, Vijayalakshmi Srinivasan, and Kailash Gopalakrishnan. PACT: parameterized clipping activation for quantized neural networks. *CoRR*, abs/1805.06085, 2018. 7

[15] Yoojin Choi, Jihwan P. Choi, Mostafa El-Khamy, and Jungwon Lee. Data-free network quantization with adversarial knowledge distillation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pages 3047–3057. IEEE, 2020. 1, 2, 3, 6, 7

[16] Yoni Choukroun, Eli Kravchik, Fan Yang, and Pavel Kisilev. Low-bit quantization of neural networks for efficient inference. In *2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, pages 3009–3018. IEEE, 2019. 2, 7

[17] Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. Binaryconnect: Training deep neural networks with binary weights during propagations. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 3123–3131, 2015. 2

[18] Tim Dettmers. 8-bit approximations for parallelism in deep learning. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016. 2

[19] Ugljesa Djuric, Gelareh Zadeh, Kenneth Aldape, and Phedias Diamandis. Precision histology: how deep learning is poised to revitalize histomorphology for personalized cancer care. *npj Precision Oncology*, 1, 12 2017. 2

[20] Carl Doersch. Tutorial on variational autoencoders. *CoRR*, abs/1606.05908, 2016. 2

[21] Ishan P. Durugkar, Ian Gemp, and Sridhar Mahadevan. Generative multi-adversarial networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. 4

[22] Alexander Finkelstein, Uri Almog, and Mark Grobman. Fighting quantization bias with bias. *CoRR*, abs/1906.03193, 2019. 2

[23] Tal Grossman. The CHIR algorithm for feed forward networks with binary weights. In David S. Touretzky, editor, *Advances in Neural Information Processing Systems 2, [NIPS Conference, Denver, Colorado, USA, November 27-30, 1989]*, pages 516–523. Morgan Kaufmann, 1989. 2

[24] Suyog Gupta, Ankur Agrawal, Kailash Gopalakrishnan, and Pritish Narayanan. Deep learning with limited numerical precision. In Francis R. Bach and David M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1737–1746. JMLR.org, 2015. 2

[25] Matan Haroush, Itay Hubara, Elad Hoffer, and Daniel Soudry. The knowledge within: Methods for data-free model compression. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 8491–8499. IEEE, 2020. 1, 2, 3, 6, 7, 8

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016. 1, 5

[27] Xiangyu He and Jian Cheng. Learning compression from limited unlabeled data. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part I*, volume 11205 of *Lecture Notes in Computer Science*, pages 778–795. Springer, 2018. 2, 6, 7

[28] Xiangyu He, Zitao Mo, Ke Cheng, Weixiang Xu, Qinghao Hu, Peisong Wang, Qingshan Liu, and Jian Cheng. Proxybnn: Learning binarized neural networks via proxy matrices. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XII*. Springer, 2020. 2

[29] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531, 2015. 2, 7

[30] Quan Hoang, Tu Dinh Nguyen, Trung Le, and Dinh Q. Phung. MGAN: training generative adversarial nets with multiple generators. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. 4

[31] Sara Hooker, Aaron Courville, Gregory Clark, Yann Dauphin, and Andrea Frome. What do compressed deep neural networks forget? arxiv e-prints, art. *arXiv preprint arXiv:1911.05248*, 2019. 4

[32] Sara Hooker, Nyalleng Moorosi, Gregory Clark, Samy Bengio, and Emily Denton. Characterising bias in compressed models. *CoRR*, abs/2010.03058, 2020. 4

[33] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 4107–4115, 2016. 2

[34] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Quantized neural networks: Training neural networks with low precision weights and activations. *J. Mach. Learn. Res.*, 18:187:1–187:30, 2017. 2, 6, 7

[35] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Francis R. Bach and David M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 448–456. JMLR.org, 2015. 1, 2, 3, 4

[36] Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew G. Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 2704–2713. IEEE Computer Society, 2018. 2, 7

[37] Sanjay Kariyappa, Atul Prakash, and Moinuddin K. Qureshi. MAZE: data-free model stealing attack using zeroth-order gradient estimation. *CoRR*, abs/2005.03161, 2020. 2

[38] Hyungjun Kim, Kyungsu Kim, Jinseok Kim, and Jae-Joon Kim. Binaryduo: Reducing gradient mismatch in binary activation network by coupling binary activations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. 2

[39] Raghuraman Krishnamoorthi. Quantizing deep convolutional networks for efficient inference: A whitepaper. *CoRR*, abs/1806.08342, 2018. 6

[40] Yann LeCun, John S. Denker, and Sara A. Solla. Optimal brain damage. In David S. Touretzky, editor, *Advances in Neural Information Processing Systems 2, [NIPS Conference, Denver, Colorado, USA, November 27-30, 1989]*, pages 598–605. Morgan Kaufmann, 1989. 5

[41] Fengfu Li and Bin Liu. Ternary weight networks. *CoRR*, abs/1605.04711, 2016. 2

[42] Xiaofan Lin, Cong Zhao, and Wei Pan. Towards accurate binary convolutional neural network. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 345–353, 2017. 2

[43] Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: differentiable architecture search. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. 5

[44] Raphael Gontijo Lopes, Stefano Fenu, and Thad Starner. Data-free knowledge distillation for deep neural networks. *CoRR*, abs/1710.07535, 2017. 2

[45] Christos Louizos, Matthias Reisser, Tijmen Blankevoort, Efstratios Gavves, and Max Welling. Relaxed quantization for discretized neural networks. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. 2

[46] Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. 5

[47] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 5188–5196. IEEE Computer Society, 2015. 1

[48] Brais Martínez, Jing Yang, Adrian Bulat, and Georgios Tzimiropoulos. Training binary neural networks with real-to-binary convolutions. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. 2

[49] Paul Micaelli and Amos J. Storkey. Zero-shot knowledge transfer via adversarial belief matching. In Hanna M.

Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 9547–9557, 2019. 2

[50] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. 7

[51] A. Mordvintsev, Christopher Olah, and M. Tyka. Inceptionism: Going deeper into neural networks. 2015. 1, 7, 8

[52] Nelson Morgan et al. Experimental determination of precision requirements for back-propagation training of artificial neural networks. In *Proc. Second Int'l. Conf. Microelectronics for Neural Networks,*, pages 9–16. Citeseer, 1991. 2

[53] Markus Nagel, Rana Ali Amjad, Mart van Baalen, Christos Louizos, and Tijmen Blankevoort. Up or down? adaptive rounding for post-training quantization. *CoRR*, abs/2004.10568, 2020. 2, 7

[54] Markus Nagel, Mart van Baalen, Tijmen Blankevoort, and Max Welling. Data-free quantization through weight equalization and bias correction. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 1325–1334. IEEE, 2019. 7

[55] Gaurav Kumar Nayak, Konda Reddy Mopuri, Vaisakh Shaj, Venkatesh Babu Radhakrishnan, and Anirban Chakraborty. Zero-shot knowledge distillation in deep networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 4743–4751. PMLR, 2019. 2

[56] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 2020. https://distill.pub/2020/circuits/zoom-in. 1

[57] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2017. https://distill.pub/2017/feature-visualization. 1

[58] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 8024–8035, 2019. 6

[59] Jorn W. T. Peters and Max Welling. Probabilistic binary neural networks. *CoRR*, abs/1809.03368, 2018. 6

[60] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part IV*, pages 525–542, 2016. 2

[61] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 91–99, 2015. 1

[62] Tim Salimans, Ian J. Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 2226–2234, 2016. 5, 8

[63] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. How good is my gan? In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part II*, volume 11206 of *Lecture Notes in Computer Science*, pages 218–234. Springer, 2018. 7

[64] Paul Smolensky. Grammar-based connectionist approaches to language. *Cogn. Sci.*, 23(4):589–613, 1999. 5

[65] Wonyong Sung, Sungho Shin, and Kyuyeon Hwang. Resiliency of deep neural networks under quantization. *CoRR*, abs/1511.06488, 2015. 7

[66] Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. Deep image prior. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 9446–9454. IEEE Computer Society, 2018. 1, 2

[67] Peisong Wang, Xiangyu He, Qiang Chen, Anda Cheng, Qingshan Liu, and Jian Cheng. Unsupervised network quantization via fixed-point factorization. *IEEE Transactions on Neural Networks and Learning Systems*, 2020. 2, 7

[68] Yulong Wang, Hang Su, Bo Zhang, and Xiaolin Hu. Interpret neural networks by identifying critical data routing paths. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 8906–8914. IEEE Computer Society, 2018. 5

[69] Shoukai Xu, Haokun Li, Bohan Zhuang, Jing Liu, Jiezhang Cao, Chuangrun Liang, and Mingkui Tan. Generative low-bitwidth data free quantization. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XII*, volume 12357 of *Lecture Notes in Computer Science*, pages 1–17. Springer, 2020. 1, 3, 6, 7

[70] 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 deep-inversion. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 8712–8721. IEEE, 2020. 1, 2, 3, 7, 8

[71] Jaemin Yoo, Minyong Cho, Taebum Kim, and U Kang. Knowledge extraction with no observable data. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 2701–2710, 2019. 2

[72] Han Zhang, Ian J. Goodfellow, Dimitris N. Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 7354–7363. PMLR, 2019. 6, 7

[73] Ritchie Zhao, Yuwei Hu, Jordan Dotzel, Christopher De Sa, and Zhiru Zhang. Improving neural network quantization without retraining using outlier channel splitting. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 7543–7552. PMLR, 2019. 2, 7

[74] Aojun Zhou, Anbang Yao, Yiwen Guo, Lin Xu, and Yurong Chen. Incremental network quantization: Towards lossless cnns with low-precision weights. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. 2

[75] Shuchang Zhou, Zekun Ni, Xinyu Zhou, He Wen, Yuxin Wu, and Yuheng Zou. Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients. *CoRR*, abs/1606.06160, 2016. 2, 7

[76] Chenzhuo Zhu, Song Han, Huizi Mao, and William J. Dally. Trained ternary quantization. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. 2