Data-Free Network Compression via Parametric Non-uniform Mixed Precision Quantization
Supplementary Material

Vladimir Chikin
Huawei Noah's Ark Lab
vladimir.chikin@huawei.com

Mikhail Antiukh
HSE University
mikhail.antiukh@gmail.com

A. Implementation of rounding on non-uniform grid

The use of cycles when performing the rounding procedure on a non-inoform grid can significantly slow down the compression procedure, so we are motivated to propose a formula for rounding on such grid based on tensor operations without using cycles and other brute force. In this section we describe an effective method of rounding numbers to proposed non-uniform grids, which we use in our paper. Let x be the full-precision value of the component of the scaled weight tensor, p be the parameter of proposed non-uniform quantization, n be the bitwidth and d be the parameter, which is expressed in terms of p and n. Let x be rounded by a positive grid point $\lfloor x \rceil_p^n$ with index k:

$$\lfloor x \rceil_p^n = d \cdot (1 + p + \cdots + p^k) = d \cdot \frac{p^{k+1} - 1}{p - 1}. \quad \text{(A.1)}$$

Motivated by Eq. (A.1), we propose the following way to calculate the value of k:

$$k = \left\lfloor \log_p \left(\frac{x \cdot (p-1)}{d} + 1 \right) \right\rfloor - 1. \quad \text{(A.2)}$$

A similar formula can be written for negative values. Formula from Eq. (A.2) defines rounding, which is slightly different from rounding to the nearest point of a non-unoform grid due to the use of a logarithm. However, this method of rounding allows us to achieve high results in compression tasks, and at the same time allows for effective implementations, so we use it in our experiments.

B. Comparison of different norms of quantization error

We have tested various variants of norms that can be used as the quantization error norm during tuning of proposed non-uniform quantization grids. We compare the results of our data-free PNMQ algorithm depending on the choice of used norm for ResNet-18, ResNet-50 and MobileNet-v2 models on ImageNet and different compression ratios from 6 to 9. Using L_m norm, we denote the traditional m-norm of a tensor divided by the number of elements in it:

$$L_m(x) = \frac{\sqrt[m]{x_1^m + \cdots + x_N^m}}{N}, \quad \text{(B.3)}$$

where $x = (x_1, \ldots, x_N) \in \mathbb{R}^N$. We compare L_1, L_2, L_4, L_6 and L_8 norms. You can see the results in Tab. C.1. As you can see, as a rule, the best results are shown by L_4 norm, and with an increase in the compression ratio, the superiority of L_4 norm over the other options increases. As a result, we propose to use L_4 norm to tune a parameters of non-uniform quantization. But our recommendation is based solely on experimental results and does not have a theoretical basis.

C. Per-channel quantization

In this section we provide the results of additional experiments using per-channel quantization technique. Per-channel quantization implies the use of different quantization parameters for different channels of weight tensors of model layers. This technique usually leads to less quality drop after quantization due to a more detailed approximation of quantization, but has a worse compression ratio due to the increased number of float parameters.

In our paper, within the framework of per-channel quantization, the size of scale s is equal to the number of channels of the layer, while the non-uniform quantization parameter p remains a scalar used for the entire layer. Also, we use a single bitwidth n for different channels of the same layer. You can see results of per-channel version of PNMQ in Tab. C.2.

D. Additional output images of compressed Faster R-CNN and Mask R-CNN

In Fig. D.1 we give a few more examples of how various compression algorithms work with object recognition

Table C.1. Data-free PNMQ: comparison of different variants of norms used to optimize proposed non-uniform grids.

Model	SCR	Top-1 acc. (%) / CR				
		L_1	L_2	L_4	L_6	L_8
ResNet-18	6	48.792 / 6.73	69.114 / 6.05	69.262 / 6.06	69.234 / 6.02	68.914 / 6.02
	6.5	48.792 / 6.73	68.568 / 6.56	69.13 / 6.61	68.794 / 6.59	68.326 / 6.59
	7	46.942 / 7.16	66.976 / 7.26	68.294 / 7.05	68.61 / 7.04	67.81 / 7.03
	7.5	35.468 / 7.58	66.326 / 7.83	68.13 / 7.51	66.052 / 7.51	65.402 / 7.51
	8	29.658 / 8.02	63.554 / 8.15	67.014 / 8.31	63.97 / 8.26	62.704 / 8.26
	8.5	11.094 / 8.57	57.678 / 8.84	66.392 / 8.53	62.646 / 8.76	61.456 / 8.73
	9	6.51 / 9.24	56.706 / 9	65.63 / 9.11	60.88 / 9.01	56.942 / 9.01
ResNet-50	6	70.374 / 6	75.592 / 6	75.444 / 6.02	75.662 / 6.02	75.242 / 6.01
	6.5	63.61 / 6.51	75.198 / 6.56	75.252 / 6.54	75.028 / 6.59	74.72 / 6.55
	7	61.35 / 7.02	74.854 / 7.03	75.188 / 7.01	74.442 / 7.01	73.676 / 7.06
	7.5	60.362 / 7.7	74.55 / 7.55	74.074 / 7.5	71.652 / 7.57	73.174 / 7.51
	8	42.04 / 8.04	72.71 / 8.17	72.728 / 8.07	70.016 / 8.06	67.238 / 8.06
	8.5	24.514 / 8.53	71.066 / 8.55	71.974 / 8.58	67.998 / 8.59	64.27 / 8.53
	9	8.268 / 9.11	69.004 / 9.02	70.854 / 9.03	66.31 / 9.04	57.108 / 9.03
MobileNet-v2	6	33.466 / 6.03	70.348 / 6.13	70.802 / 6.01	70.482 / 6.13	70.02 / 6.11
	6.5	21.078 / 6.51	69.842 / 6.5	70.14 / 6.55	69.524 / 6.53	69.078 / 6.53
	7	12.856 / 7.04	67.318 / 7.11	68.704 / 7.02	68.954 / 7	66.508 / 7.11
	7.5	12.652 / 7.67	66.66 / 7.5	67.692 / 7.53	65.69 / 7.51	63.868 / 7.51
	8	16.866 / 8.03	63.362 / 8.01	64.648 / 8.11	63.062 / 8.02	60.444 / 8.02
	8.5	10.414 / 8.5	58.12 / 8.51	60.072 / 8.52	53.446 / 8.52	48.436 / 8.52
	9	3.678 / 9.01	56.156 / 9.01	58.214 / 9.12	49.996 / 9.1	46.52 / 9.04

Table C.2. Compression of ResNet-50 and MobileNet-v2 models on ImageNet. Investigation of the impact of the per-channel quantization.

Model	Method	Type	SCR	Top-1 acc.	CR without Huffman coding	CR with Huffman coding
ResNet-50	Baseline DFQ	per-tensor	all to 5 bit	32.08%	6.36	15.17
		per-channel		73.29%	6.32	8.56
	DFQ with scale tuning	per-tensor	all to 5 bit	46.60%	6.36	11.67
		per-channel		72.45%	6.32	8.32
	Data-Free PNMQ	per-tensor	6.36	75.32%	6.43	8.34
		per-channel	6.32	75.72%	6.33	7.15
	Data-Aware PNMQ	per-tensor	6.36	75.50%	6.46	7.8
		per-channel	6.32	75.74%	6.33	6.88
MobileNet-v2	Baseline DFQ	per-tensor	all to 5 bit	55.31%	6.23	8.81
		per-channel		69.51%	6.03	6.86
	DFQ with scale tuning	per-tensor	all to 5 bit	64.14%	6.23	7.98
		per-channel		69.90%	6.03	6.73
	Data-Free PNMQ	per-tensor	6.23	70.35%	6.32	7.15
		per-channel	6.03	70.69%	6.12	6.52
	Data-Aware PNMQ	per-tensor	6.23	70.64%	6.26	6.75
		per-channel	6.03	70.71%	6.25	6.56

and image segmentation models for different compression ratios. The results show the significant superiority of our PNMQ methods over Baseline DFQ with fixed bitwidth.

E. Time for compression procedure

It is worth noting that our main goal is to improve the accuracy of data-free compression, and speedup of compres-

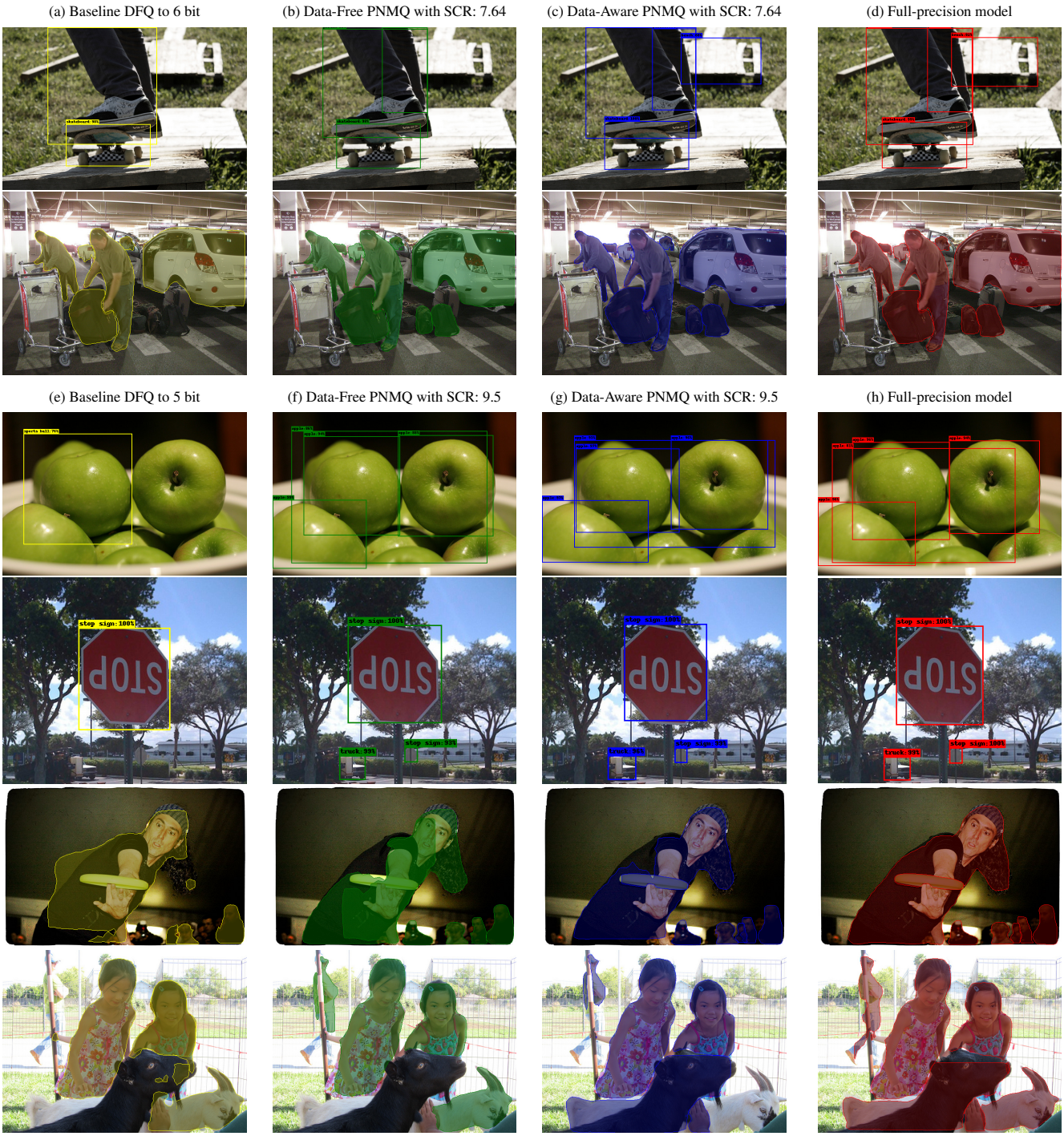

Figure D.1. Comparison of baseline DFQ method and PNMQ methods for Faster R-CNN and Mask R-CNN models on COCO-2017.

sion is not our goal. Our method can be significantly accelerated by efficient implementation and the use of faster optimization methods instead of brute force. Our PyTorch implementation is not optimal, but it allows to do compression in a short time, commensurate with baseline DFQ compression time. See this time for our implementation in Tab. E.3.

Table E.3. Time for compression for experiments from Tab. 1.

Model	Baseline DFQ	Data-Free PNMQ
ResNet-50	47.89 s	86.02 s
MobileNet-v2	47.46 s	81.32 s