CPT: EFFICIENT DEEP NEURAL NETWORK TRAINING VIA CYCLIC PRECISION

Yonggan Fu, Han Guo, Xin Yang, Yining Ding, Yingyan Lin, Meng Li & Vikas Chandra

ICLR 2021

Background

- “One forward pass of the ResNet50 model requires:
  - 4 GFLOPs (FLOPs: floating point operations) of computations
  - Full Training (on Imagenet) requires 10 FLOPs, which takes 14 days on the (then) state-of-the-art NVIDIA M40 GPU” (You et al., 2018)
- Efforts are underway to decrease the cost associated with computations. Two of the common are studying learning rates and lowering precision of networks.

Precisions:

E.g. for 8 bit: 1 bit for sign, 4 bits for mantissa and 3 bits for exponent.

→+- 1. xxxx * $2^{yyy\text{-}Max(yyy)\backslash 2}$

(Note: bits used for mantissa and exponent vary in different studies. For mentioned configuration, least positive number would be roughly 0.0176)

Existing works and proposed strategy

- Existing works:
  - have gone as low as 4-bit. But they mostly fix the model precision during the whole training process.
  - point out that a **large initial learning rate helps the model to memorize easier-to-fit and more generalizable patterns**. (Li et al., 2019)
- Novelty:
  1. Show that DNNs' precision seems to have a similar effect as the learning rate during DNN training.
  2. Proposes Cyclic Precision Training (CPT) which adopts a cyclic precision schedule along DNNs' training trajectory for pushing forward the achievable trade-offs between DNNs' accuracy and training efficiency.

- Hypothesis 1: Low precision with large quantization noise helps DNN training exploration with an effect similar to a high learning rate, while high precision with more accurate updates aids model convergence, similar to a low learning rate.

Validating Hypothesis 1. Settings: To empirically justify our hypothesis, we train ResNet-38/74 on the CIFAR-100 dataset for 160 epochs following the basic training setting as in Sec. 4.1. In particular, we divide the training of 160 epochs into three stages: [0-th, 80-th], [80-th,120-th], and [120-th, 160-th]: for the first training stage of [0-th, 80-th], we adopt different learning rates and precisions for the weights and activations, while using full precision for the remaining two stages with a learning rate of 0.01 for the [80-th,120-th] epochs and 0.001 for the [120-th, 160-th] epochs in all the experiments in order to explore the relationship between the learning rate and precision in the first training stage.

Table 1: The test accuracy of ResNet-38/74 trained on CIFAR-100 with different learning rate and precision combinations in the first stage. Note that the last two stages of all the experiments are trained with full precision and a learning rate of 0.01 and 0.001, respectively.

	ResNet-38				ResNet-74			
First-stage LR	0.1	0.06	0.03	0.01	0.1	0.06	0.03	0.01
4-bit Acc (%)	69.45	68.63	**67.69**	65.90	70.96	69.54	68.26	**67.19**
6-bit Acc (%)	70.22	68.87	67.15	**66.10**	71.62	70.28	**68.84**	66.16
8-bit Acc (%)	69.96	68.66	66.75	64.99	71.60	**70.67**	68.45	65.85
FP Acc (%)	**70.45**	**69.53**	67.47	64.50	**71.66**	70.00	68.69	65.62

- *Insights:* (1) lowering the precision introduces a similar effect of favoring exploration as that of a high learning rate; and (2) although a low precision can alleviate the accuracy drop caused by a low learning rate, a high learning rate is in general necessary to maximize the accuracy.

Validating Hypothesis 1. Settings: To empirically justify our hypothesis, we train ResNet-38/74 on the CIFAR-100 dataset for 160 epochs following the basic training setting as in Sec. 4.1. In particular, we divide the training of 160 epochs into three stages: [0-th, 80-th], [80-th,120-th], and [120-th, 160-th]: for the first training stage of [0-th, 80-th], we adopt different learning rates and precisions for the weights and activations, while using full precision for the remaining two stages with a learning rate of 0.01 for the [80-th,120-th] epochs and 0.001 for the [120-th, 160-th] epochs in all the experiments in order to explore the relationship between the learning rate and precision in the first training stage.

Table 1: The test accuracy of ResNet-38/74 trained on CIFAR-100 with different learning rate and precision combinations in the first stage. Note that the last two stages of all the experiments are trained with full precision and a learning rate of 0.01 and 0.001, respectively.

	ResNet-38				ResNet-74			
First-stage LR	0.1	0.06	0.03	0.01	0.1	0.06	0.03	0.01
4-bit Acc (%)	69.45	68.63	**67.69**	65.90	70.96	69.54	68.26	**67.19**
6-bit Acc (%)	70.22	68.87	67.15	**66.10**	71.62	70.28	**68.84**	66.16
8-bit Acc (%)	69.96	68.66	66.75	64.99	71.60	**70.67**	68.45	65.85
FP Acc (%)	**70.45**	**69.53**	67.47	64.50	**71.66**	70.00	68.69	65.62

CPT

$$B_t^n = \lceil B_{min}^n + \frac{1}{2}(B_{max}^n - B_{min}^n)(1 - cos(\frac{t\,\%\,T_n}{T_n}\pi)) \rfloor$$

B_t^n = Precision for n-th cycle and t-th epoch

N = total number of cycles

B_{max}^n, B_{max}^n are upper and lower precision bounds for nth cycle

T_n = number of training epochs in n-th cycle

N is a hyperparameter and tests show CPT is not sensitive to N. For this study, it is chosen as 32.

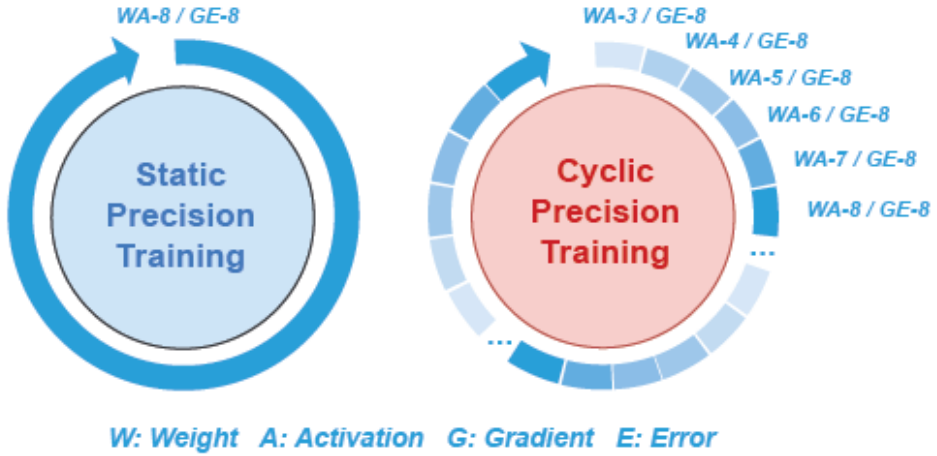


Figure 3: Static vs. Cyclic Precision Training (CPT), where CPT cyclically schedules the precision of weights and activations during training.

36.7% reduction in the required training BitOPs (bit operations), as compared to its static fixed precision counterpart.

Hypothesis 2: Dynamic precision helps DNN generalization.

- Static:
  - Weights, activations, gradients all 8 bit throughout training
- Progressive:
  - Weights and activations **progressively increased from 3 bit to 8 bit in the first 80 epochs**
  - Gradients at 8 bit
- CPT
  - Precision varied cyclically instead of progressively

CPT theorizes that this approach might better balance coarse-grained exploration and fine-grained optimization during DNN training.

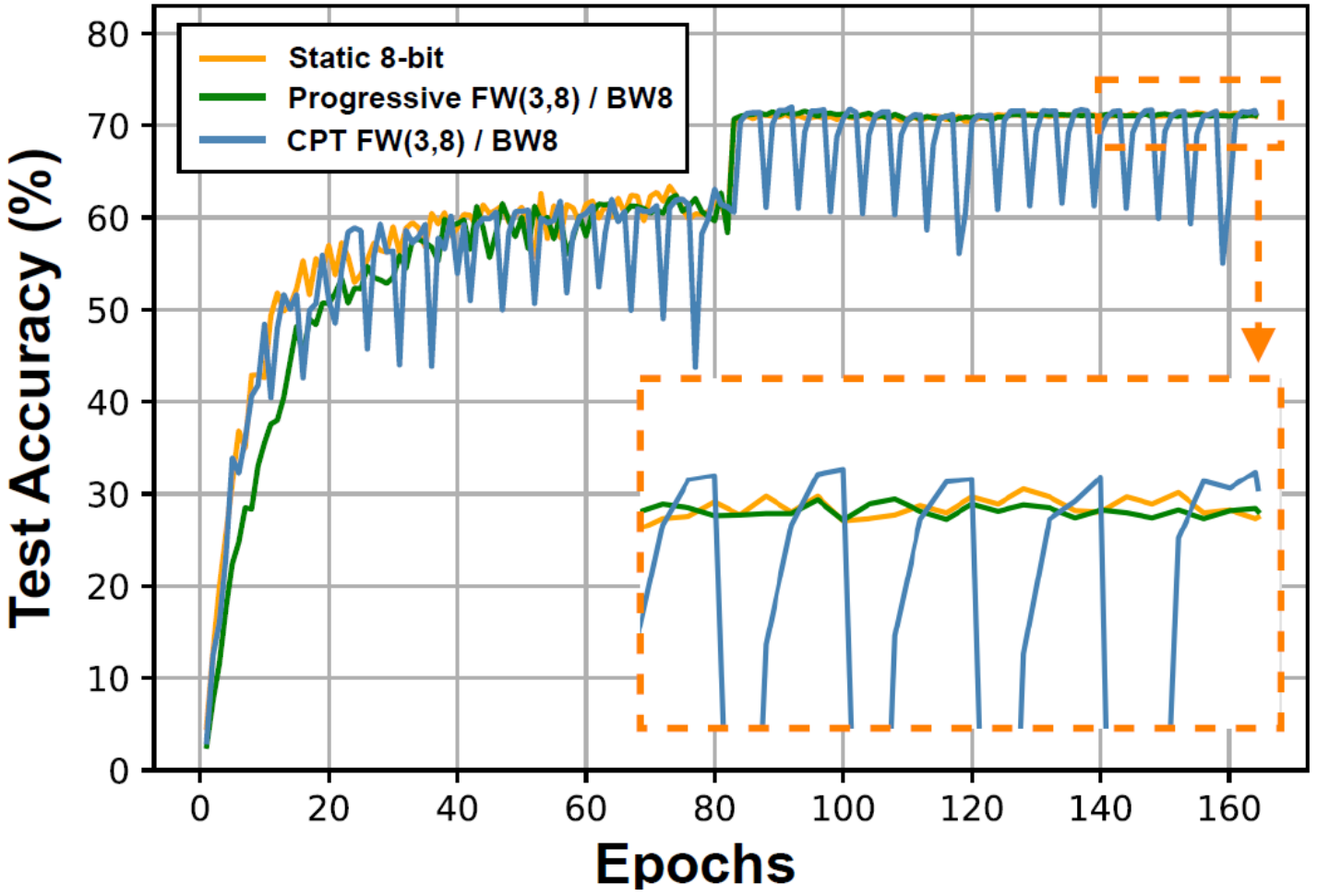
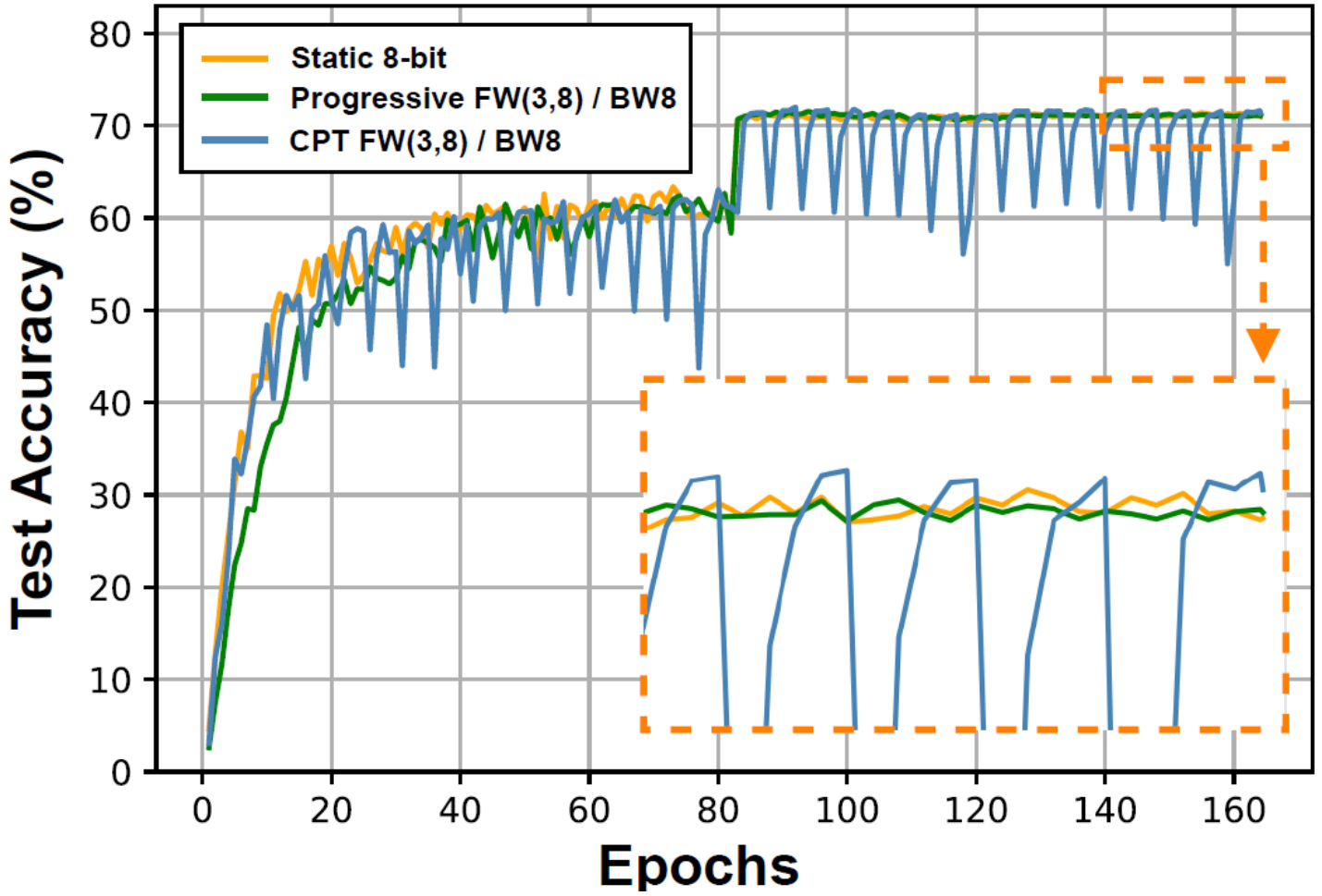


Figure 1: Test accuracy evolution of ResNet-74 on CIFAR-100 under different schedules.

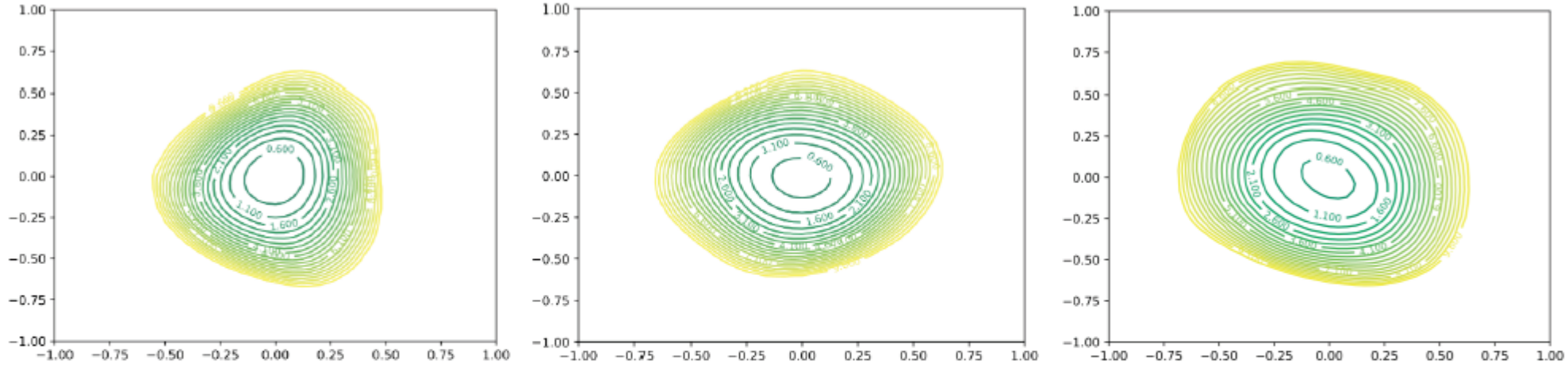
Advantages of CPT:

36.7% reduction in the required training BitOPs (bit operations), as compared to its static fixed precision counterpart.

Slightly higher accuracy (0.91%)



Loss visualization



Loss landscape of Static 8 bit (a)

Loss landscape of Progressive 3-8 bit (b)

Loss landscape of Cyclic 3-8 bit (c)

Figure 2: Loss landscape visualization after convergence of ResNet-74 on CIFAR-100 trained with different precision schedules, where wider contours with larger intervals indicate a better local minima and a lower generalization error as analyzed in (Li et al., 2018).

Visualization technique based on: Li et al (2018)

- To make contour plots: $f(\alpha, \beta) = L(\theta^* + \alpha\delta + \beta\eta)$

where θ^* is optimum parameters, alpha and beta are two hyperparameters and del and beta are two direction vectors

Contour plots are obtained by randomly sampling direction vectors from Gaussian distribution, and drawing contours through same loss values obtained.

(Filter wise normalization is done to facilitate comparison) $d_{i,j} \leftarrow \frac{d_{i,j}}{\|d_{i,j}\|}\|\theta_{i,j}\|,$

Determining precision bounds

Precision bounds are decided in first training epoch T_0, using precision range test with negligible computational overhead.

Lower bound = difference in accuracy > threshold

Upper bound = Static precision counterpart

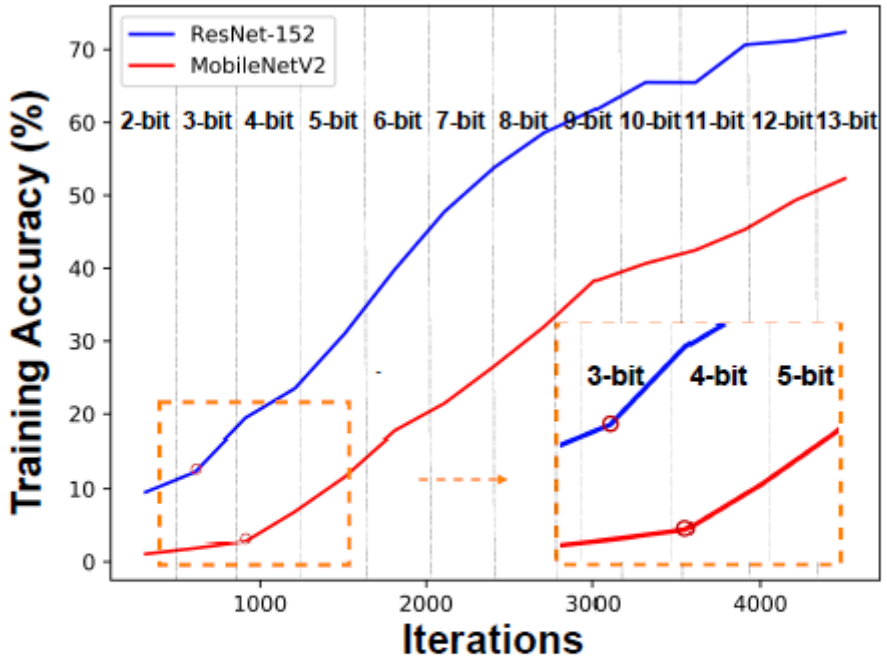


Figure 4: Illustrating the precision range test for ResNet-152 and MobileNetV2 on CIFAR-100, where the switching point which exceeds the preset threshold is denoted by red circles.

Experimental settings and results

- Eleven models:
  Eight ResNets, MobileNetv2, LSTM, Transformer
- Five Tasks:
  Cifar-10/100, Imagenet, Wikitext-103, Penn-Tree bank
- Three baselines SOTA low-precision methods in literature:
  SBM, DoReFa, WAGEUBN

Table 2: The test accuracy, computational cost, and latency of CPT, DoReFa (Zhou et al., 2016), WAGEUBN (Yang et al., 2020), and SBM (Banner et al., 2018) for training the ResNet-74/164 and MobileNetV2 models on CIFAR-10/100.

Model	Method	Precision (FW/BW)	CIFAR-10 Acc (%)	CIFAR-100 Acc (%)	GBitOPs	Latency (hour)
ResNet-74	DoReFa	8 / 8	91.16	69.31	2.67e8	44.6
	WAGEUBN	8 / 8	91.35	69.61	2.67e8	44.6
	SBM	8 / 8	92.57	71.44	2.67e8	44.6
	Proposed CPT	**3 - 8 / 8**	**93.23**	**72.35**	**1.68e8**	**35.04**
	Improv.		**+0.66**	**+0.91**	**-37.1%**	**-21.4%**
ResNet-74	DoReFa	6 / 6	90.94	69.01	1.50e8	33.2
	WAGEUBN	6 / 6	91.01	69.37	1.50e8	33.2
	SBM	6 / 6	91.15	70.31	1.50e8	33.2
	Proposed CPT	**3 - 6 / 6**	**92.4**	**70.83**	**1.05e8**	**27.5**
	Improv.		**+1.25**	**+0.52**	**-30.0%**	**-17.2%**
ResNet-164	DoReFa	8 / 8	91.40	70.90	6.04e8	101.9
	WAGEUBN	8 / 8	92.5	71.86	6.04e8	101.9
	SBM	8 / 8	93.63	72.53	6.04e8	101.9
	Proposed CPT	**3 - 8 / 8**	**93.83**	**72.9**	**3.8e8**	**80.5**
	Improv.		**+0.20**	**+0.37**	**-37.1%**	**-21.0%**
ResNet-164	DoReFa	6 / 6	91.13	70.53	3.40e8	76.7
	WAGEUBN	6 / 6	92.44	71.50	3.40e8	76.7
	SBM	6 / 6	91.95	70.34	3.40e8	76.7
	Proposed CPT	**3 - 6 / 6**	**93.02**	**71.79**	**2.37e8**	**63.5**
	Improv.		**+1.07**	**+0.29**	**-30.3%**	**-17.2%**
MobileNetV2	DoReFa	8 / 8	91.03	70.17	1.49e8	26.2
	WAGEUBN	8 / 8	92.32	71.45	1.49e8	26.2
	SBM	8 / 8	93.57	75.28	1.49e8	26.2
	Proposed CPT	**4 - 8 / 8**	**93.76**	**75.65**	**1.04e8**	**21.6**
	Improv.		**+0.19**	**+0.37**	**-30.2%**	**-17.6%**
MobileNetV2	DoReFa	6 / 6	90.25	68.4	8.39e7	18.4
	WAGEUBN	6 / 6	91.00	71.05	8.39e7	18.4
	SBM	6 / 6	91.56	72.31	8.39e7	18.4
	Proposed CPT	**4 - 6 / 6**	**91.81**	**73.18**	**6.63e7**	**15.7**
	Improv.		**+0.25**	**+0.87**	**-21.0%**	**-14.7%**

Annotation:

FW(3,8)/BW8 means CPT varied from 3-bit to 8-bit with 8-bit gradient.

Experimental settings and results

- Eleven models:
  Eight ResNets, MobileNetv2, LSTM, Transformer
- Five Tasks:
  Cifar-10/100, Imagenet, Wikitext-103, Penn-Tree bank
- Three baselines SOTA low-precision methods in literature:
  SBM, DoReFa, WAGEUBN

Table 5: The test accuracy and computational cost of (1) Transformer on WikiText-103 and (2) 2-LSTM (two-layer LSTM) on PTB, trained with CPT and SBM (Banner et al., 2018).

Model / Dataset	Method	Precision (FW/BW)	Perplexity	GBitOPs	Precision	Perplexity	GBitOPs
Transformer WikiText-103	SBM	8 / 8	31.77	1.44e6	6 / 8	32.41	9.87e5
	Proposed CPT	**4 - 8 / 8**	**30.22**	**1.0e6**	**4 - 6 / 8**	**31.0**	**7.66e5**
	Improv.		**-1.55**	**-30.2%**		**-1.41**	**-22.4%**
2-LSTM PTB	SBM	8 / 8	96.95	4.03e3	6 / 8	97.47	2.77e3
	Proposed CPT	**5 - 8 / 8**	**96.39**	**3.09e3**	**5 - 6 / 8**	**97.0**	**2.48e3**
	Improv.		**-0.56**	**-23.2%**		**-0.47**	**-10.5%**

Annotation:
FW(3,8)/BW8 means CPT varied from 3-bit to 8-bit with 8-bit gradient.

GigaBit-operations when using CPT

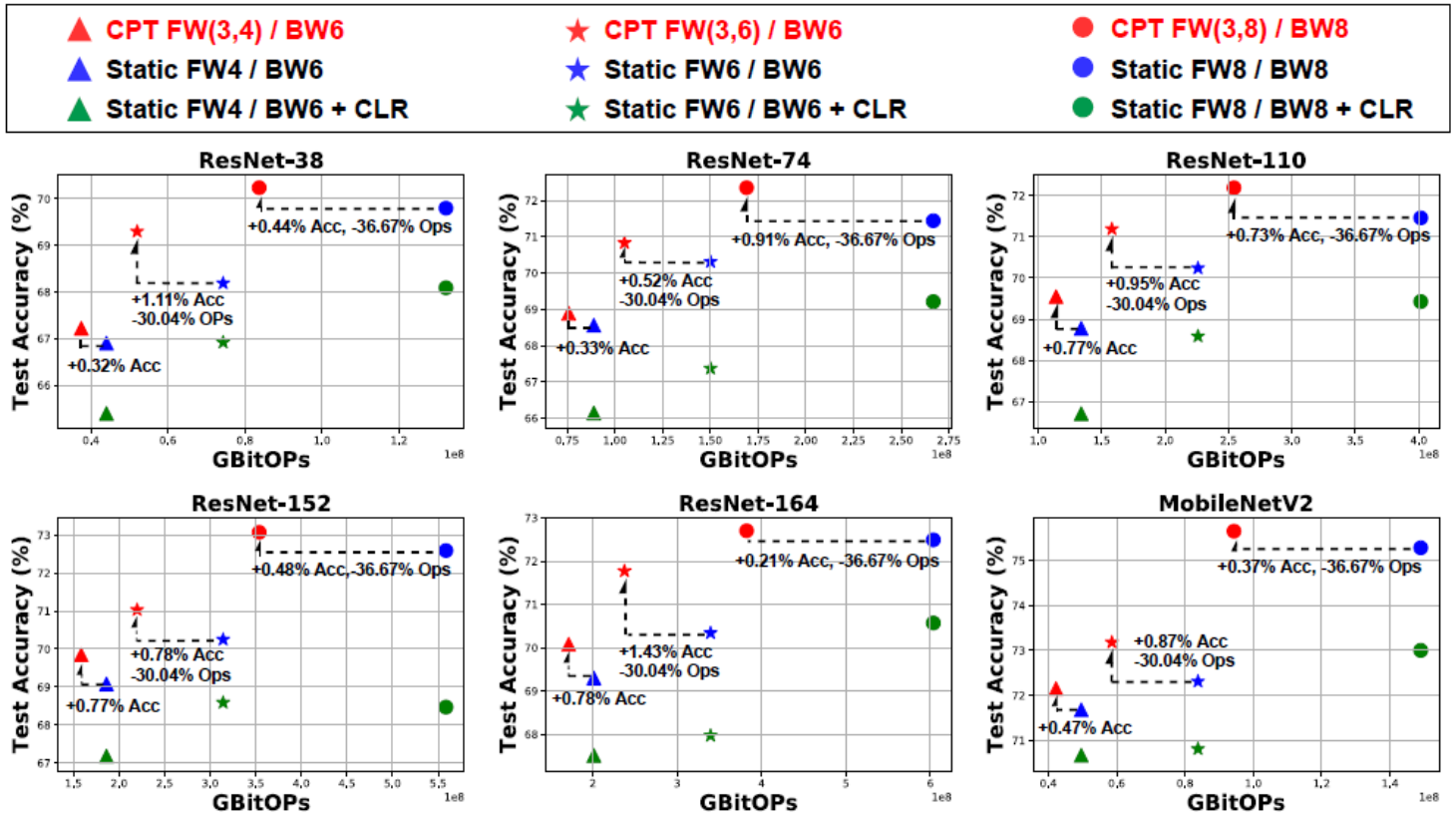


Figure 5: Test accuracy vs. the required GBitOPs when training ResNet-38/74/110/152/164 and MobileNetV2 on CIFAR-100 using static precision, static precision plus CLR, and CPT methods.

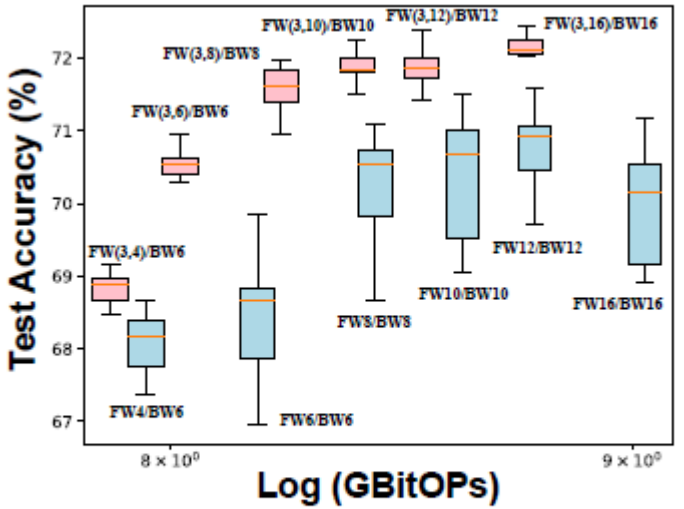


Figure 7: Training ResNet-74 on CIFAR-100 with CPT and its static counterpart.

Challenges

- Developed and tested on a custom Xilinx development board called ZC706 (Xilinx). Performance of baselines on this model might not have been well studied
- Tensorflow-lite is capable of as low as 8 bit fixed-point computation. Frameworks to implement lower than that are not mainstream yet.