[R1] Methods clarification. Sorry for not having made it clear enough. ShiftAddNet adopt SOTA bitwise-shift-based and add-based networks' design to do backpropagation (Line 176). We appreciate your suggestions and will include detailed formulation/explanation for both the backpropagation and the "fixing shift" extension in the finial revision.

[R1] Dimensions of shift/add layers. The shift layer shares the same dimensions with the original ConvNet, followed by the add layer which adapts kernel sizes / input channels to match the reduced feature maps. Although in this way, ShiftAddNet has slightly more weights than ConvNet/AdderNet (∼1.3MB for ShiftAddNet with ResNet20 (FP32) vs. 1.03 MB in the corresponding ConvNet/AdderNet, which can be further quantized to 0.4 MB without hurting the accuracy) , it takes less energy costs to achieve similar accuracies (Sec. 4.2). Since data movement is the cost bottleneck in network training/inference as **R2** mentioned (also see Tab. 1), ShiftAddNet makes an important positive step.

[R1, R2] Evaluation on two popular IoT datasets. Following your kind suggestion, we evaluate DCNN [Jiang et al. MM'15] on the popular MHEALTH [Banos et al. IWAAL'14] and USCHAD [Zhang et al. UbiComp'12] IoT benchmarks. As shown in Fig. 1 (a) and (b), ShiftAddNet again consistently outperforms the baselines under all settings in terms of accuracy-cost tradeoffs: (1) **over AdderNet**: ShiftAddNet reduces 32.8% ∼ 90.6% energy costs while resulting comparable accuracies (-0.65% ∼ 9.87%); and (2) **over DeepShift**: ShiftAddNet achieves 7.85% ∼ 30.7% higher accuracies while requiring 44.1% ∼ 74.7% less energy costs.

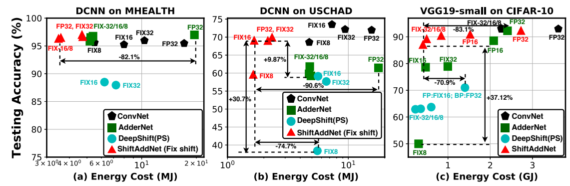

Figure 1: Accuracy vs. energy cost comparison.

[R2,R3] Larger models. As **R1** kindly mentioned, we mainly claim for energy efficiency benefit in edge computing using ResNet/VGG on CIFAR/IoT, which are popular benchmarks widely used in latest efficient CNN training papers. Furthermore, as requested we try larger models and datasets (ResNet-18/34 on ImageNet): ShiftAddNet (63.1% / 68.3%) using ResNet-18/34 architectures improves up to 3.4% top-1 accuracy than AdderNet (59.7% / 64.8%) and DeepShift (63.2% / 68.1%), with slightly higher energy overheads. Due to limited time, we train both AdderNet and ShiftAddNet with less epochs & larger batch sizes (fair comparison), after privately consulting AdderNet authors.

[R2] Completely apple-to-apple comparison. Thank you for pointing out this and providing references. We follow your advice to apply quantization training for both AdderNet and DeepShift, and compare ShiftAddNet with them in an apple-to-apple manner: Evaluated on VGG-19 with CIFAR-10 (see Fig. 1 (c)), ShiftAddNet consistently (1) improves accuracuies by 11.6%, 10.6%, 37.1% as compared to AdderNet in terms of FIX-32/16/8 formats, with comparable energy costs (-25.2% ∼ 15.7%); and (2) improves accuracies by 26.8%, 26.2%, 24.2% as compared to DeepShift (PS) in terms of FIX-32/16/8 formats, with comparable or slighly higher energy overheads. Such advantages (robustness for quantization) can also generalize to other model and dataset pairs, and we will report all of them in the final revision.

[R2] Training with FPGA. FPGA is gaining increasing population for both research (e.g., FPGA-based training framework [W. Zhao. ASAP'16]) and next-generation industrial AI (e.g., Intel FPGA acceleration [E. Chung. MICRO'18]).

[R2] Wider comparisons using ASIC&FPGA. We follow your suggestion to supply a comprehensive comparison using both ASIC and FPGA, and analyze the energy savings from both the operation and model perspectives (see Tab. 1). Addition and bit-wise shift help to save 1.1× ∼ 7.6× and 3.8× ∼ 9.9× energy costs over multiplication based ConvNet, respectively, where the FPGA energy is measured on board and ASIC energy costs are measured using a SOTA predictor [Xu et al. FPGA'20].

Table 1:Wider comparisons using ASIC&FPGA.

Inference Type	**Format**		**ASIC (45nm)**		**FPGA**	
	Operation	Format	Energy (pJ)	Improv.	Energy (pJ)	Improv.
Operation energy	Mult.	FP32	3.7	-	18.8	-
		FIX32	3.1	-	19.6	-
		FIX8	0.2	-	0.2	-
	Add	FP32	0.9	4.1x	0.4	47x
		FIX32	0.1	31x	0.1	196x
		FIX8	0.03	6.7x	0.1	2x
	Shift	FIX32	0.13	24x	0.1	196x
		FIX8	0.024	8.3x	0.025	8x
	Operation	Format	Energy (MJ)	Improv.	Energy (GJ)	Improv.
Model energy (VGG-19 small)	Mult.	FP32	8.08	-	3.6	-
		FIX32	7.35	-	2.27	-
		FIX8	1.54	-	2.27	-
	Add	FP32	6.17	1.3x	2.4	1.5x
		FIX32	5.76	1.3x	1	2.3x
		FIX8	1.44	1.1x	0.3	7.6x
	Shift	FIX32	0.87	8.5x	0.6	3.8x
		FIX8	0.21	7.3x	0.23	9.9x

[R3] ❶ FPGA measurement: We measure the dynamic power (by power meter) and latency for one iteration, and then scale the energy costs to the whole training process; **❷ Hardware area and throughput:** We by default ensure the hardware cost (area) approximately the same for all: a default frequency of 100MHz and a throughput of 13FPS / 20FPS for FIX-32/8 using ResNet-20 on CIFAR; **❸ Inference costs:** E.g., when training DCNN on IoT dataset (see Fig. 1 (a)), ShiftAddNet (FIX-32; fix shift) costs 1.7 J, where AdderNet (FIX-32) costs 1.9 J and DeepShift (FIX-32) costs 2.6 J), respectively, leading to 10.5% / 34.6% savings; **❹ FPGA energy breakdown:** E.g., Clocks: 7%, Signals: 6%, Logic: 5%, BRAM: 10%, DSP: 1%, PS7: 71%, for ShiftAddNet (FIX-8) with ResNet-20 on CIFAR; **❺ Complete comparisons:** We supply the additional cases as you suggested, e.g., when training VGG-19 on CIFAR-10 (see Fig. 1 (c)), ShiftAddNet reduces -25.3% ∼ 83.1% energy costs over AdderNet, while offering comparable accuracies (-5.17% ∼ 37.12%), and meanwhile achieves 16.1% ∼ 24.2% higher accuracies, while reducing -43.6% ∼ 70.9% energy costs over DeepShift; **❻ Comparable energy costs (line 202):** It precisely means ConvNet costs ±30% more than AdderNet (FP32); **❼ Mixed quantization:** We follow [Elthakeb et al. MICRO'20] to try mixed precision training methods for ShiftAddNet (Acc.: 88.5%) vs. 88.2% with FIX-32, energy: 28.8% savings over FIX-32; **❽ Fig. 4 reference:** Sorry for the missing reference, we will add it in Sec. 4.4.1. We appreciate all of these questions and promise to supply experiments on all the above settings and over E^2Train in the final revision.