To All Reviewers: With positive scores from all the reviewers, we thank all the reviewers for their valuable feedbacks. Code has been submitted as the supplementary material and will be publicly released if the paper gets accepted.

Q: Justification for self-adjustment of Eq. (11). We design Eq. (11), $\tilde{\mathbf{w}}^i = \mathbf{w}^i + \left((\mathbf{R}^i)^T\mathbf{w}^i - \mathbf{w}^i\right)\cdot\alpha^i$ to further mitigate angular bias. In Sec. 3.3, we adopt alternating optimization for the non-convex objective of Eq. (7). It does reduce the angular bias, which, however, cannot guarantee the global optimum. Usually, it either overshoots (Fig. A(a)) or undershoots (Fig. A(b)) the binarization $\text{sign}\left((\mathbf{R}^i)^T\mathbf{w}^i\right)$. Eq. (11) constrains that the final weight vector moves along the residual direction of $(\mathbf{R}^i)^T\mathbf{w}^i - \mathbf{w}^i$ with $\alpha \geq 0$. It is intuitive that when overshooting, $\alpha \leq 1$; when undershooting, $\alpha \geq 1$. We empirically observe that overshooting is in a dominant position. Thus, we simply constrain $\alpha \in [0, 1]$ to shrink the learnable domain of α, which we find can well further reduce the quantization error and boost the performance as demonstrated in Tab. 4 of the paper. The final value of α varies across different layers. In Fig. A(c), we show how α updates during training in ResNet-20 (layer2.2.conv2). We will clarify this part in our final version.

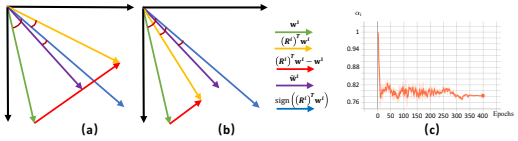

Figure A: Viewed with zooming in.

Reviewer #1: clear accept (8) – insightful, interesting, attractive and pleasure to read.

Q1: Per-iteration rotation. Taking ResNet-20 on CIFAR-10 (one GTX 1080 GPU) as an example, per-iteration rotation achieves 86.7% (53.33 hours) top-1 accuracy while per-epoch rotation achieves 86.5% (5.83 hours). Per-epoch rotation achieves 9.15× speed-up with negligible accuracy compromise. Thus we adopt per-epoch rotation. **Q2: Fig. 2(b) and Fig. 5.** Fig. 2(b) shows per-layer quantization error consisting of λ, $\mathbf{w}$ and $\mathbf{b}$ in Eq. (1), and Fig. 5 shows per-layer distribution of $\mathbf{w}$. Since λ was ignored in Fig. 5, it is inappropriate to compute quantization error by comparing $\mathbf{w}$ with their centers (binarization). **Q3: Angular bias of activation.** It consumes additional computation in the test stage since the activation is input-related. Thus, we do not apply activation rotation. **Q4: Typos.** They have been fixed.

Reviewer #2: above the acceptance (6) – new, interesting, clearly written and sufficiently explained.

Q1: Approximation brings more improvements. Our approximation achieves a good accuracy of 86.6% based on a mediocre baseline XNOR-Net (83.7%), which is hard to be further boosted. Nevertheless, our rotation still increases it to 87.1%. Actually, combining our rotation with XNOR-Net (without approximation) can also obtain a great performance of 86.4%. Thus, our rotation should not be evaluated simply by a relative increment from 86.6% to 87.1%. **Q2: Additional parameters.** Additional parameters include α^i, $\mathbf{R}_1^i$ and $\mathbf{R}_2^i$, which are floating-point. α^i brings negligible consumption. Theoretical complexities of $\mathbf{R}_1^i$ and $\mathbf{R}_2^i$ are provided in line 127 of the paper. We take ResNet-20 on CIFAR-10 to test actual complexity. It takes 21 (19) seconds to finish one training epoch on one GTX 1080 GPU with (without) bi-rotation. Besides, only 0.06MB space are introduced additionally. Both the time and space cost are negligible. Moreover, $\mathbf{R}_1^i$ and $\mathbf{R}_2^i$ are applied in the training stage to learn weight binarization, which means no overhead in the test stage. **Q3: Approximation motivation.** The approximation has to be "soft" to backward gradient, and also has to well approximate the sign function to minimize the forward error. Though $\tanh$ function enables gradient propagation, it never reaches the exact values of -1/+1 even with annealing hyperparameter. Our approach not only enables gradient propagation, but also behaves exactly the same as the sign function in the ends which distinguishes our method from others. In our final version, we will reorganize Sec. 3.4 and include more related works including $\tanh$ (annealing). **Q4: Penalizing the angle while learning.** Though reducing the angle error, we find it hard to increase the probability of weight flip. Thus, the rotation is introduced. **Q5: loss of bi-rotation.** Bi-rotation doesn't increase the loss since the bi-rotation matrices are equivalent to full-matrix rotation (see line 125 of the paper and our response to **Q2** of Reviewer #3). **Q6: Test time overhead?** No (see **Q2**).

Reviewer #3: above the acceptance (6) – innovative, interesting, and good awareness of the literature.

Q1: How to compensate for rotating the weights? At the beginning of each training epoch, we apply the weight rotation to reduce the angle first. A regular training epoch is then applied on basis of the rotated weights to retain the accuracy (see lines 145–147 of the paper). **Q2: Going-forward suggestions.** We really appreciate your valuable suggestions. First, since Martinez *et al.*, haven't released their code yet, we could not verify currently if properly training the network will correct the angular bias as the rebuttal period is very limited. However, we will do it after the rebuttal. Second, our GPU platform can perform full rotation on CIFAR-10 and it shows similar results to bi-rotation. However, it exceeds our hardware capacity to evaluate on ImageNet because of the massive memory consumption. In sum, our final version will include the following changes: 1) Angular bias and weight flip will be clearly defined in the introduction. 2) The validation will be re-formatted and table notations will be explained in the caption. 3) More existing approximations will be added in Sec. 3.4. 4) Lines 37, 86 and 104 will be rephrased. 5) Eq. (11) will be justified. 6) We will rewrite our broader impact.

Reviewer #4: above the acceptance (6) – novel, interesting, well written and easy to understand.

Q1: Typos. We have carefully proofread the manuscript and fixed the typos.