SinReQ: Generalized Sinusoidal Regularization for Low-Bitwidth Deep Quantized Training

Ahmed T. Elthakeb Prannoy Pilligundla Hadi Esmaeilzadeh
Alternative Computing Technologies (ACT) Lab
University of California San Diego
{a1yousse, ppilligu, hadi}@eng.ucsd.edu

ABSTRACT

Deep quantization of neural networks (below eight bits) offers significant promise in reducing their compute and storage cost. Albeit alluring, without special techniques for training and optimization, deep quantization results in significant accuracy loss. To further mitigate this loss, we propose a novel sinusoidal regularization, called SinReQ, for deep quantized training. SinReQ adds a *periodic* term to the original objective function of the underlying training algorithm. SinReQ exploits the periodicity, *differentiability*, and the desired convexity profile in sinusoidal functions to automatically propel weights towards values that are inherently closer to quantization levels. Since, this technique does not require invasive changes to the training procedure, SinReQ can harmoniously enhance quantized training algorithms. SinReQ offers generality and flexibility as it is not limited to a certain bitwidth or a uniform assignment of bitwidths across layers. We carry out experimentation using the CIFAR-10, ResNet-20, SVHN, and VGG-11 DNNs with three to five bits for quantization and show the versatility of SinReQ in enhancing multiple quantized training algorithms, DoReFa [1] and WRPN [2]. Averaging across all the bit configurations shows that SinReQ closes the accuracy gap between these two techniques and the full-precision runs by 35.7% and 37.1%, respectively. That is improving the absolute accuracy of DoReFa and WRPN up to 5.3% and 2.6%, respectively.

1. INTRODUCTION

Despite the success of DNNs in various domains [3, 4, 5], their compute efficiency hinders effective deployment in resource-limited platforms [6]. Quantization, in general, and deep quantization, in particular, aim to not only reduce the compute requirements of DNNs but also significantly reduce their memory footprint [1, 2, 7, 8]. Nevertheless, without specialized training and optimization training algorithms, quantization can diminish the accuracy. As such, several techniques have been proposed that aim to train DNNs in quantized mode with as low as possible loss in accuracy [9, 10, 11, 12]. However, eliminating the loss has proven to be illusive.

This paper aims to provide a new mechanism that enhances these techniques and significantly closes the remaining gap between deeply quantized and full precision networks. As such, we propose a sinusoidal regularization technique, a differentiable loss, that naturally pushes the weight values toward the quantization levels exploiting the inherent periodicity of sinusoidal functions. As such, quantized training algorithms [1, 2] that still use some form of backpropagation [13] can effectively utilize the proposed mechanism to further enhance their performance in accuracy recovery.

SinReQ offers generality and can be used with different bitwidths by setting the periodicity of the regularizer according to the desired bitwidth. Moreover, the proposed technique is flexible and a dedicated sinusoidal term for each layer with different periods can enable heterogenous quantization across the layers. The SinReQ regularization can also be applied for training a model from scratch, or for fine-tuning a pretrained model. We evaluate SinReQ using different bitwidth assignments across different DNNs (CIFAR-10, ResNet-20, SVHN, and VGG-11). To show the versatility of SinReQ, it is used with two different quantized training algorithms, DoReFa [1] and WRPN [2]. Over all the bitwidth assignments, the proposed regularization, improves the top-1 accuracy of DoReFa and WRPN up to 5.3% and 2.6%, respectively. That is, on average, closing the gap between the quantized network and a full-precision netwrok by 35.7% in the case of DoReFa and 37.1% in the case of WRPN.

2. RELATED WORK

SinReQ, which is a regularization technique, is complementary to the previously proposed quantized training [9, 10, 11, 12] and binarization [14, 15] algorithms and can potentially augment their training procedure. Additionally, there is a line of research that aims to incorporate the distance between the quantized levels and the full-precision weights in the training loss [16, 17, 18, 19, 20]. In contrast, SinReQ, utilizes the periodic nature of sinusoidal function to push the weight values to the quantization levels possible by the allocated bitwidth to each layer.

Training algorithms for quantized neural networks..
There have been several techniques [1, 21, 2] that train a neural network in a quantized domain after the bitwidth of the layers is determined manually. DoReFa quantizes weights, activations and gradients of neural networks using different

bitwidths. They suggest maintaining a high-precision floating point copy of the weights while feeding quantized weights into backprop. WRPN introduces a scheme to train networks from scratch using reduced-precision activations by decreasing the precision of both activations and weights and increasing the number of filter maps in a layer. [21] performs the training phase of the network in full precision, but for inference uses ternary weight assignments. For this assignment, the weights are quantized using two scaling factors which are learned during training phase. SinReQ is a complimentary method that can potentially enhance these algorithms. The paper demonstrates this feature concretely in the context of DoReFa and WRPN training algorithms.

Binarized and ternarized neural networks..
Extensive work, [22, 15, 23] focus on binarized neural networks, which impose accuracy loss but reduce the bitwidth to lowest possible level. In BinaryNet [14], an extreme case, a method is proposed for training binarized neural networks which reduce memory size, accesses and computation intensity at the cost of accuracy. XNOR-Net [15] leverages binary operations (such as XNOR) to approximate convolution in binarized neural networks. On the other side, in a weight ternarized network, zero is used as an additional quantized value. [23] introduces ternary-weight networks, in which the weights are quantized to -1, 0, +1 values by minimizing the Euclidian distance between full-precision weights and their ternary assigned values. In [24] different scaling factors are introduced to the ternarized weights. The scaling parameters are learned by gradient descent. None of these techniques propose sinusoidal regularization to make the weight values more quantization friendly as training progresses.

Loss-aware weight quantization..
Recent works pursued loss-aware minimization approaches for quantization. [17, 18] developed approximate solutions using proximal Newton algorithm to minimize the loss function directly under the constraints of low bitwidth weights. [19] proposed to learn the quantization of DNNs through regularization by introducing a learnable regularization coefficient to find low bitwidth models efficiently in training. [20] proposed an adaptive technique to jointly train a quantized, bit-operation-compatible DNN and its associated quantizers, as opposed to using fixed, handcrafted quantization schemes such as uniform or logarithmic quantization. Although these techniques use regularization to guide the process of quantized training, they don not explore the use of periodic differentiable trigonometric functions.

3. SINUSOIDAL REGULARIZATION FOR AUTOMATIC QUANTIZATION DURING TRAINING

Our proposed method SinReQ exploits weight regularization in order to automatically quantize a neural network while training. To that end, Sections 3.1 to 3.3 describe the role of regularization in neural networks and then Section 3.4 explains SinReQ in more detail.

3.1 Loss Landscape of Neural Networks

Neural networks' loss landscapes are known to be highly non-convex and generally very poorly understood. It has been empirically verified that loss surfaces for large neural networks have many local minima that are essentially equivalent in terms of test error [25], [26]. Moreover, converging to one of the many good local minima proves to be more useful as compared to struggling to find the global minimum of the accuracy loss on the training set (which often leads to overfitting). This opens up and encourages a possibility of adding extra custom objectives to optimize for during the training process, in addition to the original objective (i.e., the accuracy loss). The added custom objective could be with the purpose of increasing generalization performance or imposing some preference on the weights values. Regularization is one of the major techniques that makes use of such facts as discussed in the following subsection.

3.2 Regularization in Neural Networks

Neural networks often suffer from redundancy of parameterization and consequently they commonly tend to overfit. Regularization is one of the commonly used techniques to enhance generalization performance of neural networks. Regularization effectively constrains weight parameters by adding a term (regularizer) to the objective function that captures the desired constraint in a soft way. This is achieved by imposing some sort of preference on weight updates during the optimization process. As a result, regularization seamlessly leads to unconditionally constrained optimization problem instead of explicitly constrained which, in most cases, is much more difficult to solve.

3.3 Classical regularization: weight decay.

The most commonly used regularization technique is known as *weight decay*, which aims to reduce the network complexity by limiting the growth of the weights. It is realized by adding a term to the objective function that penalizes large weight values

$$E(w) = E_o(w) + \frac{1}{2}\lambda \sum_i {w_i}^2 \quad (1)$$

where E_o is the original loss measure, and λ is a parameter governing how strongly large weights are penalized. w_i is a vector of all weights for layer i of the network while summation of i is over all the layers in the network.

3.4 Periodic Regularization: SinReQ.

In this work, we propose a new type of regularization that is friendly to quantization. The proposed regularization is based on a periodic function (sinusoidal) that provides a smooth and differentiable loss to the original objective, Figure 1 (a). The periodic regularizer has a periodic pattern of minima that correspond to the desired quantization levels. Such correspondence is achieved by matching the period to the quantization step based on a particular number of bits for a given layer.

$$E(w) = E_o(w) + \frac{1}{2}\lambda \sum_i {w_i}^2 + \frac{1}{2}\lambda_q \sum_i \sin^2\left(\frac{\pi w_i}{2^{-qbits}-1}\right) \quad (2)$$

where E_o is the original loss measure, and λ_q is SinReQ regularization strength that is a parameter governing how strongly weight quantization errors are penalized. For the sake of simplicity and clarity, Figure 1 (b) and (c) depict a geometrical

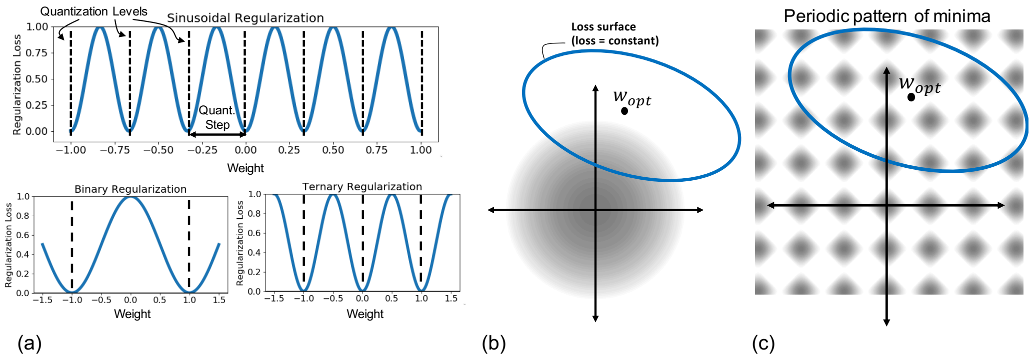

Figure 1: (a) Generalized SinReQ profile adapting for arbitrary bitwidths including binary and ternary quantization. (b) and (c) depict a geometrical sketch for a hypothetical loss surface (original objective function to be minimized) and an extra regularization term in 2-D weight space for weight decay and SinReQ respectively. w_{opt} is the optimal point just for the loss function alone.

Table 1: Summary of results comparing state-of-the-art methods DoReFa, and WRPN with and without SinReQ for different neural networks

Network	Weights Bitwidth	Top1 Accuracy (%) DoReFa	DoReFa +SinReQ	WRPN	WRPN +SinReQ	Full Precision Accuracy (%)
CIFAR10	3 bits	58.80	**63.28**	55.26	**56.46**	**74.91**
	4 bits	65.54	**70.88**	68.72	**70.84**	
	5 bits	73.12	**74.10**	73.94	**74.82**	
SVHN	3 bits	88.40	**90.2**	88.12	**90.13**	**97.12**
	4 bits	93.81	**94.70**	93.22	**94.31**	
	5 bits	96.76	**96.89**	96.51	**96.87**	
ResNet-20 on CIFAR10	3 bits	87.08	**88.7**	76.22	**78.39**	**93.53**
	4 bits	90.16	**91.20**	82.10	**84.72**	
	5 bits	90.58	**91.82**	89.08	**91.67**	

sketch for a hypothetical loss surface (original objective function to be minimized) and an extra regularization term in 2-D weight space. For weight decay regularization, in Figure 1 (b), the faded circular contours show that as we get closer to the origin, the regularization loss is minimized. w_{opt} is the optimum just for the loss function alone and the overall optimum solution is achieved by striking a balance between the original loss term and the regularization loss term.

In a similar vein, Figure 1 (c) shows a representation of the proposed regularization. A periodic pattern of minima pockets are seen surrounding the original optimum point. The objective of the optimization problem is to find the best solution that is the closest to one of those minima pockets where weight values are nearly matching the desired quantization levels, hence the name quantization-friendly. Algorithm 1 details the implementation procedure of SinReQ regularization using LeNet as an example.

Algorithm 1 SinReQ implementation on LeNet

$qbits \leftarrow$ number of quantization bits; $qbits \in \{1,2,3,...\}$
$\lambda_q \leftarrow$ regularization strength
▷ Set the quantization step based on the used quantization technique
▷ for DoReFa quantization
$step \leftarrow 1/(2^{qbits} - 0.5)\,, \Delta \leftarrow step/2$
▷ for WRPN quantization
$step \leftarrow 1/(2^{qbits} - 1.0)\,, \Delta \leftarrow 0$
▷ For each layer in the network, calculate the sinreq loss
▷ Layer conv1
$kernel \leftarrow conv1.float\_weight$
$sinreq_{cv1} \leftarrow reduce\_mean(sin^2(\pi \times (kernel + \Delta))/step)$
▷ Layer conv2
$kernel \leftarrow conv2.float\_weight$
$sinreq_{cv2} \leftarrow reduce\_mean(sin^2(\pi \times (kernel + \Delta))/step)$
▷ Layer fc1
$kernel \leftarrow fc1.float\_weight$
$sinreq_{fc1} \leftarrow reduce\_mean(sin^2(\pi \times (kernel + \Delta))/step)$
▷ Layer fc2
$kernel \leftarrow fc2.float\_weight$
$sinreq_{fc2} \leftarrow reduce\_mean(sin^2(\pi \times (kernel + \Delta))/step)$
▷ Sum over all layers
$sinreq\_loss = sinreq_{cv1} + sinreq_{cv2} + sinreq_{fc1} + sinreq_{fc2}$
▷ Calculate the overall loss
$LOSS = original\_loss + \lambda_q \times sinreq\_loss$

4. EVALUATION: SINREQ IN ACTION

To demonstrate the effectiveness of our proposed sinusoidal regularization, we evaluated it on three neural networks (CIFAR10, SVHN, and ResNet-20) Here, we focus on fine-tuning from a pretrained models as compared to training from scratch.

4.1 Experimental Setup.

We implemented our technique inside Distiller [27], an open source framework for compression by Intel Nervana. The reported accuracies for DoReFa and WRPN are with the built-in implementations in Distiller, which may not exactly match the accuracies reported in their respective papers. However, an independent implementation from a major company provides an unbiased foundation for the comparisons.

4.2 Semi-quantized weight distributions.

Figure 2 shows the evolution of weights distributions over fine-tuning epochs for different layers of (a) CIFAR10 and (b) SVHN networks at different bitwidths (3, 4, and 5 bits). The high-precision weights form clusters and gradually converge

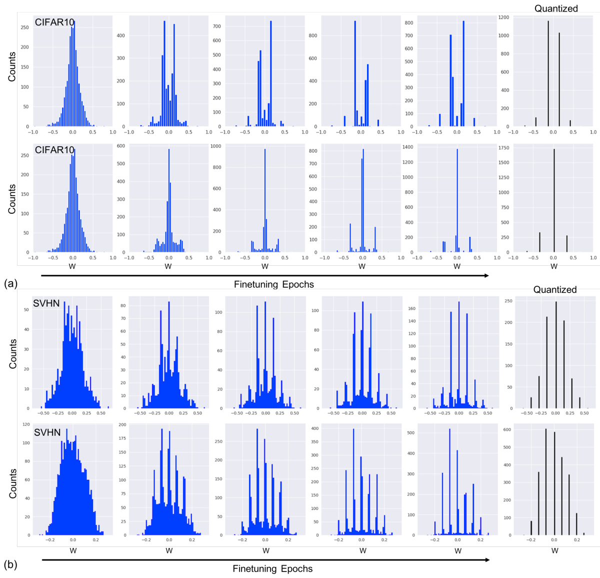

Figure 2: Evolution of weight distributions over training epochs (with the proposed regularization) at different layers and bitwidths for CIFAR10 and SVHN.(a) CIFAR10, second convolution layer with 3 bits, top row: mid-rise type of quantization (shifting by half a step to exclude zero as a quantization level); bottom row: mid-tread type of quantization (zero is included as a quantization level). (b) SVHN, top row: first convolution layer with 4 bits; bottom row: first fully connected layer with 5 bits.

around the quantization centroids (i.e., levels) as regularization loss is minimized along with the main accuracy loss. The rate of convergence to the target quantization levels depends on (i) the number of fine-tuning epochs, (2) the regularization strength (λ_q). It is worth noting that λ_q is a hyper-parameter that controls the tradeoff between the accuracy loss and the regularization loss. Fixed value can be presumed ahead of training or fine-tuning, however careful setting of such parameter can yield optimum results. [19] considers the regularization coefficient as a learnable parameter.

4.3 Arbitrary-bitwidth quantization.

Considering the following sinusoidal regularizer, with $step_q$ denoting the quantization step, and Δ is an offset.

$$R(W) = \lambda_q \sum_i \sin^2\left(\frac{\pi w_i + \Delta}{step_q}\right) \tag{3}$$

SinReQ provides generality in two aspects. First, the flexibility to adapt for arbitrary number of bits. The parameter $step_q$ controls the periodicity of the sinusoidal function. Thus, for any arbitrary bitwidth ($qbits$), $step_q$ can be tuned to match the respective quantization step. For uniform quantization:

$$step_q = 2^{-qbits} - 1$$

The second aspect of generality is the seamless accommodation for different quantization styles. There are two styles of uniform quantization: mid-tread and mid-rise. In mid-tread, zero is considered as a quantization level, while in mid-rise, quantization levels are shifted by half a step such that zero is not included. Ternary quantization, using $\{-1,0,1\}$, is an example of the former, while binary quantization, using $\{-1,1\}$, is an example of the latter. Figure 2 (a) shows the second conv layer of CIFAR10 at 3 bits, top row: mid-rise quantization, and bottom row: mid-tread quantization.

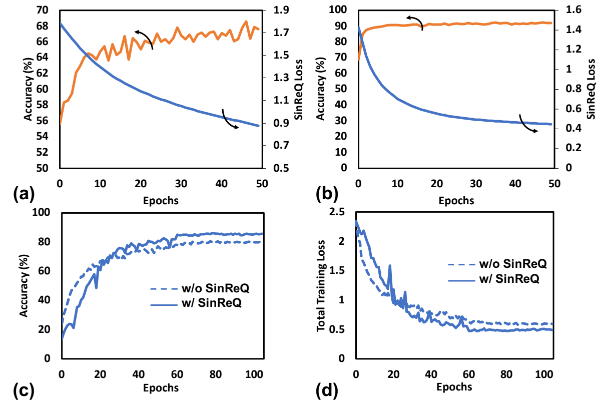

Figure 3: Convergence behavior: accuracy and SinReQ regularization loss over fine-tuning epochs for (a) CIFAR10, (b) SVHN. Comparing convergence behavior with and without SinReQ during training from scratch (a) accuracy, (b) training loss

4.4 Layer-wise optimization.

As different layers have different levels of sensitivity to the quantization bitwidth [28], enabling layer-wise quantization opens the possibility for heterogenous bitwidth quantization and consequently more optimized quantized networks. This can be achieved by adding a custom regularizer (as shown in equation 3) for each layer and sum over all layers. Then, we add the regularization losses of all layers to the main accuracy loss and pass the entire collective loss to the gradient-descent optimizer.

4.5 Comparison to existing methods.

We assess the efficacy of SinReQ on boosting the performance of existing methods for training quantized networks, DoReFa, and WRPN. Table 1 summarizes the accuracies obtained by DoReFa, and WRPN with and without SinReQ. Results show that integrating SinReQ within the training algorithm achieves up to 5.3%, and 2.6% accuracy improvements to DoReFa, and WRPN methods respectively. That is averaging to around 2% and 1.6% respectively.

4.6 Convergence analysis.

Figure 3 (a), and (b) show the convergence behavior of SinReQ by visualizing both accuracy and regularization loss over finetuning epochs for two networks: CIFAR10 and SVHN. As can be seen, the regularization loss (SinReQ Loss) is minimized across the finetuning epochs while the accuracy is maximized. This demonstrates a validity for the proposed regularization being able to optimize the two objectives simultaneously. Figure 3 (c), and (d) contrasts the convergence behavior with and without SinReQ for the case of training from scratch for VGG-11. It can be noticed that, at the onset of training, the accuracy in the presence of SinReQ is behind that without SinReQ. This can be explained as a result of optimizing for an extra objective in case of with SinReQ as compared to without. Shortly thereafter, the regularization effect kicks in and eventually achieves $\sim$6% accuracy improvement. The convergence behavior, however, is primarily controlled by the regularization strength (λ_q). As briefly mentioned in section

3.4, $\lambda_q \in [0, \infty)$ is a hyperparameter that weights the relative contribution of the proposed regularization objective to the standard accuracy objective. In the context of neural networks, it is sometimes desirable to use a separate setting of λ_q for each layer of the network. Throughout our experiments, λ_q is set the same across all layers and in the range of $0.5-10$. We reckon that careful setting of λ_q across the layers and during the training epochs is essential for optimum results [19].

5. CONCLUSION

Deep quantization of DNNs promises to be a powerful technique in reducing their complexity. However, it comes with the vice of loss in accuracy that needs to be remedied. This paper provided a new approach in using sinusoidal regularizations terms to push the weight values closer to the quantized levels. This mathematical approach is versatile and augments other quantized training algorithms by improving the quality of the network they train. While this technique consistently improves the accuracy, SinReQ does not require changes to the base training algorithm or the neural network topology.

6. REFERENCES

[1] S. Zhou, Z. Ni, X. Zhou, H. Wen, Y. Wu, and Y. Zou, "DoReFa-Net: Training Low Bitwidth Convolutional Neural Networks with Low Bitwidth Gradients," *CoRR*, 2016.

[2] A. K. Mishra, E. Nurvitadhi, J. J. Cook, and D. Marr, "WRPN: Wide Reduced-Precision Networks," in *ICLR*, 2018.

[3] Y. LeCun, Y. Bengio, and G. E. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[4] J. Hauswald, M. Laurenzano, Y. Zhang, C. Li, A. Rovinski, A. Khurana, R. G. Dreslinski, T. N. Mudge, V. Petrucci, L. Tang, and J. Mars, "Sirius: An open end-to-end voice and vision personal assistant and its implications for future warehouse scale computers," in *ASPLOS*, 2015.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States.* (P. L. Bartlett, F. C. N. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), pp. 1106–1114, 2012.

[6] V. Sze, Y. Chen, T. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2295–2329, 2017.

[7] H. Sharma, J. Park, N. Suda, L. Lai, B. Chau, V. Chandra, and H. Esmaeilzadeh, "Bit fusion: Bit-level dynamically composable architecture for accelerating deep neural network," *2018 ACM/IEEE 45th Annual International Symposium on Computer Architecture (ISCA)*, pp. 764–775, 2018.

[8] P. Judd, J. Albericio, T. H. Hetherington, T. M. Aamodt, and A. Moshovos, "Stripes: Bit-serial deep neural network computing," *2016 49th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, pp. 1–12, 2016.

[9] M. Courbariaux, Y. Bengio, and J. 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* (C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, eds.), pp. 3123–3131, 2015.

[10] S. Gupta, A. Agrawal, K. Gopalakrishnan, and P. Narayanan, "Deep learning with limited numerical precision," in *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015* (F. R. Bach and D. M. Blei, eds.), vol. 37 of *JMLR Workshop and Conference Proceedings*, pp. 1737–1746, JMLR.org, 2015.

[11] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Quantized neural networks: Training neural networks with low precision weights and activations," *Journal of Machine Learning Research*, vol. 18, pp. 187:1–187:30, 2017.

[12] A. Zhou, A. Yao, Y. Guo, L. Xu, and Y. 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* [29].

[13] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations* (D. E. Rumelhart and J. L. Mcclelland, eds.), pp. 318–362, Cambridge, MA: MIT Press, 1986.

[14] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized Neural Networks," in *Advances in Neural Information Processing Systems 29* (D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, eds.), 2016.

[15] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks," in *ECCV*, 2016.

[16] S. Jung, C. Son, S. Lee, J. Son, Y. Kwak, J. Han, and C. Choi, "Joint training of low-precision neural network with quantization interval parameters," *CoRR*, vol. abs/1808.05779, 2018.

[17] L. Hou, Q. Yao, and J. T. Kwok, "Loss-aware binarization of deep networks," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings* [29].

[18] L. Hou and J. T. Kwok, "Loss-aware weight quantization of deep networks," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, OpenReview.net, 2018.

[19] Y. Choi, M. El-Khamy, and J. Lee, "Learning low precision deep neural networks through regularization," *CoRR*, vol. abs/1809.00095, 2018.

[20] D. Zhang, J. Yang, D. Ye, and G. Hua, "Lq-nets: Learned quantization for highly accurate and compact deep neural networks," in *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VIII* (V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, eds.), vol. 11212 of *Lecture Notes in Computer Science*, pp. 373–390, Springer, 2018.

[21] C. Zhu, S. Han, H. Mao, and W. J. Dally, "Trained Ternary Quantization," in *ICLR*, 2017.

[22] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Quantized Neural Networks: Training Neural Networks with Low Precision Weights and Activations," *J. Mach. Learn. Res.*, 2017.

[23] F. Li and B. Liu, "Ternary Weight Networks," *CoRR*, vol. abs/1605.04711, 2016.

[24] C. Zhu, S. Han, H. Mao, and W. J. Dally, "Trained ternary quantization," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings* [29].

[25] A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous, and Y. LeCun, "The Loss Surfaces of Multilayer Networks," in *Artificial Intelligence and Statistics*, 2015.

[26] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the Loss Landscape of Neural Nets," in *Advances in Neural Information Processing Systems*, 2018.

[27] N. Zmora, G. Jacob, and G. Novik, "Neural network distiller," June 2018.

[28] A. T. Elthakeb, P. Pilligundla, A. Yazdanbakhsh, F. Mireshghallah, and H. Esmaeilzadeh, "Releq: A reinforcement learning approach for deep quantization of neural networks," *CoRR*, vol. abs/1811.01704, 2018.

[29] *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, OpenReview.net, 2017.