

BD-Net: Has Depth-Wise Convolution Ever Been Applied in Binary Neural Networks?

DoYoung Kim, Jin-Seop Lee, Noo-ri Kim, SungJoon Lee, Jee-Hyong Lee*

Sungkyunkwan University, Suwon, South Korea
{kacel33, wlstjq0602, pd99j, sjoon8379, john}@skku.edu

Abstract

Recent advances in model compression have highlighted the potential of low-bit precision techniques, with Binary Neural Networks (BNNs) attracting attention for their extreme efficiency. However, extreme quantization in BNNs limits representational capacity and destabilizes training, posing significant challenges for lightweight architectures with depth-wise convolutions. To address this, we propose a 1.58-bit convolution to enhance expressiveness and a pre-BN residual connection to stabilize optimization by improving the Hessian condition number. These innovations enable, to the best of our knowledge, the first successful binarization of depth-wise convolutions in BNNs. Our method achieves 33M OPs on ImageNet with MobileNet V1, establishing a new state-of-the-art in BNNs by outperforming prior methods with comparable OPs. Moreover, it consistently outperforms existing methods across various datasets, including CIFAR-10, CIFAR-100, STL-10, Tiny ImageNet, and Oxford Flowers 102, with accuracy improvements of up to 9.3 percentage points.

Code — <https://github.com/kacel33/BD-Net>

Introduction

Deep Neural Networks (DNNs) have demonstrated significant performance improvements in various tasks, including image classification (He et al. 2016; Rawat and Wang 2017; Kim, Lee, and Lee 2024), object detection (Redmon et al. 2016; Zou et al. 2023), and semantic segmentation (Long, Shelhamer, and Darrell 2015; Minaee et al. 2021). Despite this success, they require substantial parameters and computational cost for training and inference. Consequently, most DNNs require computational accelerators like high-end GPUs, making them unsuitable for edge devices in resource-constrained environments. Recently, there has been a growing demand for personalized on-device services using DNNs, highlighting the importance of developing lightweight models that can operate efficiently in these limited environments. To achieve lightweight models, model compression methods have been proposed, such as knowledge distillation (Hinton, Vinyals, and Dean 2015; Gou et al. 2021), pruning (Han et al. 2015; Liang et al. 2021), compact

*Corresponding author

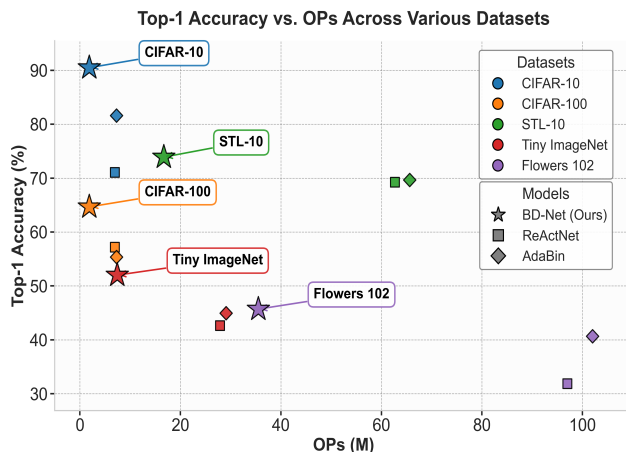


Figure 1: Top-1 accuracy vs. operations (millions of OPs) across various datasets. Colors denote datasets: CIFAR-10 (blue), CIFAR-100 (orange), STL-10 (green), Tiny ImageNet (red), and Flowers-102 (purple). Marker shapes represent models: BD-Net (stars), ReActNet (squares), and AdaBin (diamonds). A blue star, for example, indicates BD-Net applied to CIFAR-10. BD-Net consistently achieves superior accuracy with significantly fewer operations across all datasets.

networks (Howard et al. 2019; Zhang et al. 2018; Han et al. 2020), and low-bit precision (Zhou et al. 2016; Jacob et al. 2018; Rakka et al. 2022; Shin et al. 2023).

Recently, Binary Neural Networks (BNNs) have attracted significant attention because they can drastically reduce computational costs by quantizing both model weights and activations to 1-bit. Since they use 1-bit operations, traditional multiplication-accumulation operations can be replaced by XNOR-bitcount operations, making them highly effective in resource-constrained environments like low-power CPUs. The key challenge in BNNs is minimizing performance degradation. Since weights and activations are quantized to 1-bit precision, the network’s representational capacity is significantly reduced, leading to performance degradation. To address this issue, quantization approximations (Rastegari et al. 2016; Lin, Zhao, and Pan 2017; Tu et al. 2022) and adaptive thresholds (Liu et al. 2020; Lee

et al. 2023) have been proposed to reduce quantization errors.

However, these binarization approaches have proven effective only for DNNs with regular convolutions and have not been successfully applied to compact architectures such as the MobileNet series (Howard et al. 2017; Sandler et al. 2018; Howard et al. 2019). Attempts to binarize depth-wise convolutions (DW convs) using existing approaches result in substantial performance degradation. Although DW convs are essential components for lightweight model architectures, current binarization techniques fail to effectively quantize these operations. To address these limitations of BNNs and enhance their applicability in creating more lightweight models, new specialized binarization methods specifically designed for depth-wise separable convolutions are needed.

The difficulty in effectively binarizing DW convs arises from a combination of two factors: the use of 1-bit precision and the structural limitation of DW convs. Usually, low-bit precision quantization techniques control the model performance by adjusting the number of bits for quantization (Gluska and Grobman 2020; Wang et al. 2019). However, in BNNs, which use only 1-bit precision, handling quantization errors becomes particularly challenging. The structural characteristic of DW convs is the small number of parameters compared to regular convs. Due to this, it is challenging for quantization errors to be effectively mitigated (Finkelstein, Almog, and Grobman 2019). Additionally, regardless of quantization, due to the channel-wise independent convolution operation, DW convs often exhibit zero variances in batch normalization (BN) layers, leading to large gradient fluctuations (Sheng et al. 2018). When this structural feature is combined with binarization, gradients may fluctuate more, resulting in a very rough loss landscape, and model training becomes very challenging.

We propose a method for effectively **binarizing DW conv operations in BNNs** while minimizing performance loss. To reduce quantization errors in BNNs, we introduce 1.58-bit conv operations to enhance representational capacity and a pre-BN residual connection to smooth the loss landscape, leading to more stable training. These techniques enable the first successful binarization of DW convs in BNNs. As shown in Figure 1, our method significantly reduces OPs while simultaneously improving accuracy across multiple datasets, achieving accuracy gains of up to 9.3 percentage points (p.p.) on CIFAR-10, CIFAR-100, STL-10, Tiny ImageNet, and Oxford Flowers 102. On ImageNet, it maintains competitive accuracy with approximately 3 times fewer OPs compared to regular conv-based BNNs. This represents a significant reduction in computational cost compared to current state-of-the-art BNNs.

Preliminaries

In BNNs, the computational cost can be reduced by replacing traditional multiplication-accumulation operations with XNOR-bitcount operations. The total operations (OP) are determined by both binary operations (BOP) and floating point operations ($FLOP$), which are expressed as $OP = BOP/64 + FLOP$ (Bulat and Tzimiropoulos 2019; Liu

Type	Operation	Count
Full-precision	3×3 regular conv.	462M
	3×3 depth-wise conv.	3.61M
Binary	3×3 regular conv.	7.23M
	3×3 depth-wise conv.	56K

Table 1: Operation count comparison of convolutional operations for full-precision and binary networks. The counts are for a CNN layer where the resolution is 56×56 , and both the input and output channels are 128, respectively.

et al. 2018). However, despite the significant compression ratio, conventional BNNs do not significantly reduce computational cost when binarizing 32-bit compact networks. This is mainly because DW conv has not been successfully binarized. In this section, we delve into the reasons why DW conv needs to be binarized, and explore the challenges that have prevented its successful binarization.

Why depth-wise convolution needs to be binarized

The depth-wise separable structure introduced in the MobileNet series is a key technology for lightweight CNN models. In this structure, DW conv requires significantly less computation compared to regular convs.

Table 1 provides a comparison of different conv operations along with their operation counts of a CNN layer. Interestingly, binary regular convs still require more computation than 32-bit DW conv. While BNNs are more efficient than standard 32-bit models, they are not necessarily more efficient than compact models using 32-bit DW convs.

We can binarize DW conv similarly to regular convs. However, such a simple attempt frequently results in decreased performance. For this reason, early BNN research (Liu et al. 2020; Tu et al. 2022; Lee et al. 2023) replaced 3×3 DW conv in MobileNet V1’s backbone with 3×3 regular convs. While this substitution improved performance, it did not significantly reduce computational cost. Theoretically, BNN approaches could reduce model size by up to 64 times, but ReActNet (Liu et al. 2020) achieved only a 6.5-fold reduction, from 569M OPs in full-precision MobileNet V1 to 87M OPs. We performed experiments on ReActNet with DW conv and observed a substantial drop in performance, reducing Top-1 accuracy on ImageNet from 69.4% to 60.7%. Similar issues have also been observed in low-bit precision environments. In mixed precision settings, more bits are usually allocated in depth-wise separable convolutions (Gluska and Grobman 2020; Wang et al. 2019; Sheng et al. 2018). However, in BNNs, we cannot allocate more bits because we use only 1-bit precision.

Therefore, to build more efficient and lightweight models in BNNs, it is crucial to effectively binarize DW conv.

Why depth-wise convolution hasn’t been binarized

Problems caused by quantization are closely related to the number of bits and the number of parameters (Finkelstein, Almog, and Grobman 2019). If a model is quantized with

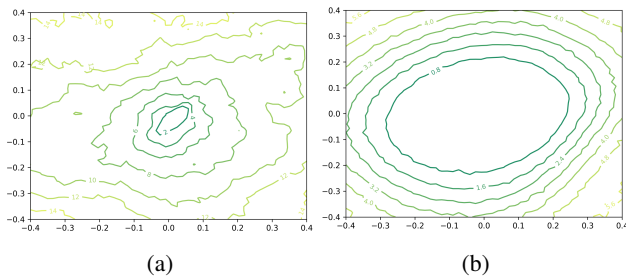


Figure 2: Loss landscapes of (a) binarized MobileNet V1 and (b) full-precision MobileNet V1.

a small number of bits, the reduced representational capacity leads to lower model performance. Additionally, using a small number of quantization bits can cause the weights or activation values to fluctuate significantly near the quantization rounding boundaries during training. Such rapid changes lead to sudden changes in gradients, making the learning process unstable and slowing down convergence speed (Nagel et al. 2022).

However, when the number of parameters in a model is large, both the quantization errors from each parameter and the changes in weights and activations can be easily canceled out. As a result, the influence of quantization errors decreases, and the gradients become relatively stable, facilitating the model’s learning.

This is why previous studies have successfully binarized regular convs, but have not binarized DW conv, which has fewer learnable parameters. Particularly in BNNs, the use of 1-bit quantization results in very high quantization errors, causing the gradients to change drastically. This makes it more challenging for models with fewer parameters to handle these effects properly.

Independent of quantization, DW conv has another problem due to its structural characteristics, which causes learning instability. In regular convs, each output channel is influenced by all input channels, but in DW conv, each channel receives only one input channel. Therefore, the outputs of a channel are likely to have small variances (Sheng et al. 2018). The output of a BN layer can be described as follows:

$$y = \alpha(x - \mu) + \beta, \text{ where } \alpha = \frac{\gamma}{\sqrt{\sigma^2 + \epsilon}}. \quad (1)$$

Here, y is the normalized output, γ and β are the scale and shift parameters, and α is the scaling factor. If σ approaches zero, the scaling factor α can become abnormally large. It leads to large gradients, making the model training process potentially unstable (Sheng et al. 2018).

The difficulties in binarizing DW conv can be summarized as follows:

- Due to fewer learning parameters, depth-wise convolution struggles to mitigate quantization errors, lowering the model’s accuracy and increasing learning instability.
- The 1-bit quantization, with only two values to represent, not only reduces the model performance, but also significantly increases learning instability due to extreme changes in weights during training.

- Depth-wise convolution is structurally more prone to learning instability. If it is binarized, the instability can be amplified.

These three factors make binarized DW convs not only highly inaccurate, but also extremely difficult to train due to severe learning instability.

Figure 2 illustrates the loss landscapes of binarized MobileNet V1 and full-precision MobileNet V1. Despite the structural issues of DW conv, the full-precision version shows a relatively smooth landscape due to its full precision representation. However, the binarized MobileNet V1, affected by all the three factors, displays a very rough landscape. To successfully binarize DW conv with high performances, methods that can enhance expressiveness with only 1-bit quantization and smooth the loss landscape for training stability are required.

Related Work

Binary Neural Networks (BNNs) quantize both activations and weights to 1-bit, replacing traditional floating-point operations with faster bitwise operations. While this significantly accelerates computation, the reduction from 32-bit to 1-bit precision leads to information loss and resulting in performance degradation. To address these challenges and narrow the performance gap between BNNs and full-precision networks, methods such as quantization approximation (Rastegari et al. 2016; Lin, Zhao, and Pan 2017; Tu et al. 2022), adaptive thresholding (Liu et al. 2020; Lee et al. 2023) and structural modification (Liu et al. 2018; Ma et al. 2024) have been proposed.

In quantization approximation, XNOR-Net (Rastegari et al. 2016) introduces a real-valued scaling factor to improve approximation accuracy and ABC-Net (Lin, Zhao, and Pan 2017) simulates full-precision behavior using linear combinations of multiple binary convs. AdaBin (Tu et al. 2022) replaces fixed binary sets with adaptively obtained optimal binary sets. Additionally, adaptive thresholding techniques further reduce quantization errors. ReActNet (Liu et al. 2020) introduces learnable parameters in the binary sign function with RSign and RReLU, enabling dynamic adjustment during training. INSTA-BNN (Lee et al. 2023) applies instance-aware thresholding using a statistical measure for more precise binarization. In terms of structural modifications, Bi-Real Net (Liu et al. 2018) preserves information through residual connections, while A&B BNN (Ma et al. 2024) eliminates 32-bit multiplication in BNNs with a removable mask layer and quantized RReLU.

Despite these advances, binarizing DW convs has been particularly challenging due to structural characteristics that lead to training instability. This has resulted in performance drops and minimal FLOP savings compared to full-precision compact networks. To avoid the training instability of binarizing DW convs, MoBiNet (Phan et al. 2020a) and Binary MobileNet (Phan et al. 2020b) used group convs instead. However, they still have not fully leveraged the computational efficiency of DW convs.

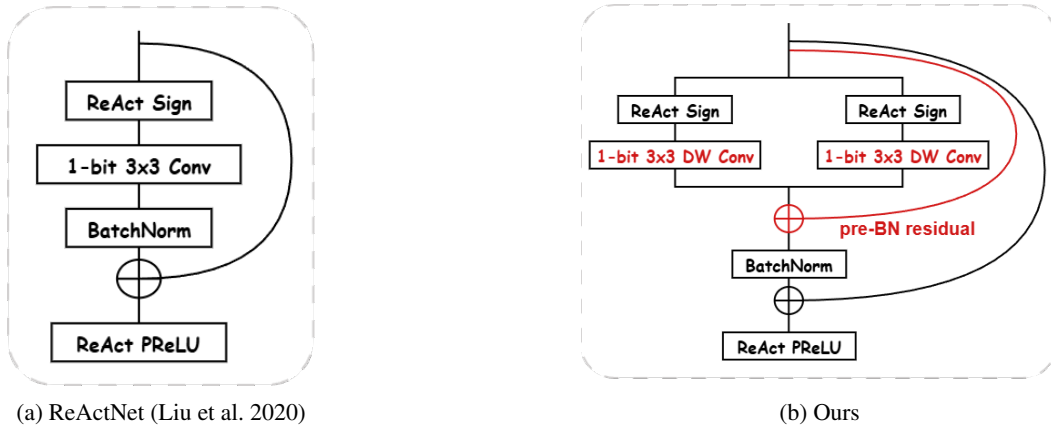


Figure 3: Architecture comparison of binary depth-wise convolution layers.

Methodology

In this section, we propose methods to effectively binarize DW convs while maintaining high performance. We first introduce the pre-BN residual structure, which stabilizes the training of binary DW convs, followed by the 1.58-bit technique, which enhances their representational capacity.

Pre-BN residual connection on binary DW conv

Binarizing DW convs is challenging due to structural constraints and extremely low bit widths. While prior studies have proposed residual strategies to stabilize binary networks, these approaches have been limited to binary regular convs. To the best of our knowledge, binary DW convs have not been studied before, and their severe ill-conditioning problems remain unaddressed. In this work, we identify these challenges and introduce the pre-BN residual connection, a simple yet effective method designed specifically to alleviate the ill-conditioning of binary DW convs. As illustrated in Figure 3, this connection directly links the input of the binary DW conv to the input of the subsequent BN layer, smoothing the loss landscape and stabilizing optimization, all without introducing additional learnable parameters.

A rough loss landscape is closely associated with a high condition number of the Hessian matrix (Marteau-Ferey, Bach, and Rudi 2019). The condition number of the Hessian matrix is defined as follows:

$$\kappa(H) = \|H\| \|H^{-1}\| = \frac{\lambda_1(H)}{\lambda_n(H)} \quad (2)$$

where $\lambda_1(H)$ and $\lambda_n(H)$ denote the largest and smallest eigenvalues of H , respectively. A higher condition number indicates greater sensitivity to perturbations and a more rugged loss landscape.

Let J be the Jacobian of a naive binary DW conv with BN, and J' be that of our BD-Net with the pre-BN residual. Their corresponding Hessians are denoted as H and H' , respectively. The Jacobian J can be expressed as:

$$J = \alpha J^{dw} + I \quad (3)$$

where α is the scaling factor of the BN as defined in Equation (1), and J^{dw} is the Jacobian of the DW conv layer. The

Jacobian of BD-Net is then formulated as:

$$J' = \alpha J^{dw} + (\alpha + 1)I. \quad (4)$$

Since the condition number of the Hessian approximately satisfies $\kappa(H) \approx \kappa(J)^2$, the condition numbers of H and H' can be approximated as:

$$\kappa(H) \approx \left\{ \frac{\lambda_1(J)}{\lambda_n(J)} \right\}^2, \quad (5)$$

$$\kappa(H') \approx \left\{ \frac{\lambda_1(J')}{\lambda_n(J')} \right\}^2 = \left\{ \frac{\lambda_1(J) + \alpha}{\lambda_n(J) + \alpha} \right\}^2. \quad (6)$$

The condition number of J' is computed as:

$$\begin{aligned} \kappa(J') &= \frac{\lambda_1(J')}{\lambda_n(J')} = \frac{\lambda_1(J) + \alpha}{\lambda_n(J) + \alpha} \\ &= \frac{\lambda_1(J) + \alpha}{\lambda_n(J) + \alpha} \cdot \frac{\lambda_n(J)}{\lambda_1(J)} \cdot \frac{\lambda_1(J)}{\lambda_n(J)} \\ &= \frac{\lambda_n(J)(\lambda_1(J) + \alpha)}{\lambda_1(J)(\lambda_n(J) + \alpha)} \cdot \frac{\lambda_1(J)}{\lambda_n(J)} \\ &= \left(1 + \frac{\alpha}{\lambda_1(J)} \right) \cdot \left(\frac{\lambda_n(J)}{\lambda_n(J) + \alpha} \right) \cdot \frac{\lambda_1(J)}{\lambda_n(J)}. \end{aligned} \quad (7)$$

In binary DW convs, the variance of the output is often near zero due to extreme quantization, which causes the BN scaling factor α to become significantly large. In such cases, since $\lambda_n(J) \ll \alpha$, we can approximate:

$$\kappa(J') \approx \left(1 + \frac{\alpha}{\lambda_1(J)} \right) \cdot \frac{\lambda_n(J)}{\alpha} \cdot \frac{\lambda_1(J)}{\lambda_n(J)}. \quad (8)$$

Consequently, the condition number of J' becomes:

$$\kappa(J') \approx \left(\frac{\lambda_n(J)}{\alpha} + \frac{\lambda_n(J)}{\lambda_1(J)} \right) \cdot \kappa(J). \quad (9)$$

As α increases, the condition number of the DW conv layer with the pre-BN residual approaches $\frac{\lambda_n(J)}{\lambda_1(J)} \cdot \kappa(J)$, leading to a significant reduction. Therefore, the condition numbers satisfy:

$$\kappa(H') < \kappa(H). \quad (10)$$

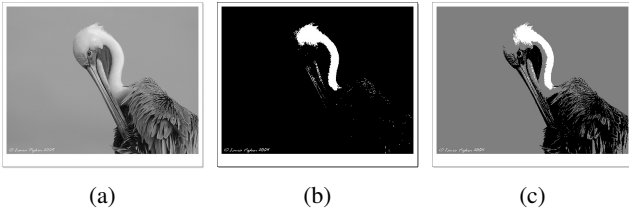


Figure 4: Image visualization on ImageNet validation dataset. (a) Converted to grayscale, (b) Binarized using Otsu’s algorithm, (c) Displayed using an appropriate threshold to represent in 1.58-bit.

This demonstrates that the pre-BN residual systematically controls the condition number via α , directly flattening the loss landscape and ensuring stable convergence during training. This structural solution effectively mitigates the limitations of binary DW convs with minimal computational overhead with no additional learnable parameters. The practical benefits of the pre-BN residual are further validated in the Experiments section through extensive experiments, which demonstrate notable improvements in both training stability and accuracy.

1.58-bit convolution

DW conv has been successfully implemented in full-precision, contributing to the lightweight design of CNN-based models. However, unlike 32-bit DW conv, binary DW conv suffers from critically limited filter diversity. For example, while binary regular conv can generate $2^{9 \times C_{in}}$ filter combinations, binary DW conv is restricted to only 2^9 combinations of $\{-1, 1\}$ filters. This limitation significantly reduces the network’s ability to capture complex features, leading to poor representational capacity.

To address this limitation, we propose a dual binary conv structure that connects two binary convs in parallel and sums their outputs. Unlike prior work such as ABC-Net (Lin, Zhao, and Pan 2017), which improves performance through linear combinations of multiple binary convs at the cost of excessive computational overhead (up to 25 times more convs), our method requires only a small increase in computation while significantly enhancing representational power.

In our approach, each binary conv has its own rounding boundary (α) and output value ($\pm\beta$), and the final output is determined by combining the results of two binary convs. When N binary convs are used, the number of output combinations becomes $N + 1$. If we define the effective bit precision as $M = \log_2(N + 1)$, the representational effect of the output grows logarithmically with respect to N . For example, when $N = 2$, we obtain $M = \log_2(3) \approx 1.58$, which inspires the term “1.58-bit conv.”

$$f(x) = \begin{cases} -\beta_1 - \beta_2 & \text{if } x < \alpha_1, \\ \beta_1 - \beta_2 & \text{if } \alpha_1 \leq x < \alpha_2, \\ \beta_1 + \beta_2 & \text{if } \alpha_2 \leq x. \end{cases} \quad (11)$$

It is important to clarify that 1.58-bit equivalent does not refer to the actual bitwidth of weights or activations used

in the network. Instead, it reflects the increased representational capacity achieved by combining multiple binary convs while maintaining bitwise operations. In this sense, the term expresses the enhanced expressive power of the dual binary conv rather than the physical storage or computational bit precision.

While $N = 3$ would yield an effect similar to 2-bit precision, using more than two binary convs introduces greater computational cost without proportional benefits. Therefore, we adopt the dual binary conv ($N = 2$) as an efficient trade-off between computation and representational capacity.

This method preserves the inherent efficiency of bitwise operations in BNNs while improving expressiveness. As illustrated in Figure 4, conventional BNNs with a single rounding boundary may suffer from severe distortions in the shape and color representation of objects due to aggressive binarization. In contrast, the 1.58-bit equivalent conv alleviates such distortions to some extent, providing improved representational capacity while maintaining the efficiency of binary operations.

Experiments

We primarily aim to explore the feasibility and potential benefits of incorporating DW convs into BNNs. To achieve this, we conduct experiments on CIFAR-10, CIFAR-100 (Krizhevsky and Hinton 2009), STL-10 (Coates, Ng, and Lee 2011), Tiny ImageNet (Yang 2015), Oxford Flowers 102 (Nilsback and Zisserman 2008), and ILSVRC12 ImageNet (Russakovsky et al. 2015).

To demonstrate the broader applicability of our approach, we implement our components on both ReActNet and AdaBin architectures, confirming the effectiveness of our method across different network designs. BD-Net-A (ReActNet) and BD-Net-A (AdaBin) modify ReActNet and AdaBin, respectively, by replacing the 3×3 regular convs with 3×3 DW convs and incorporating Pre-BN residual connections and 1.58-bit convs. Additionally, we introduce BD-Net-B (ReActNet) and BD-Net-B (AdaBin), variants of our method that replaces 1-bit 3×3 DW convs in downsampling layers with real-valued 3×3 DW convs in ReActNet and AdaBin, respectively. For a fair comparison, the first and last layers are kept in full precision in all cases.

Experimental Results

To demonstrate the effectiveness and applicability of our proposed method, we conduct experiments on small- and medium-scale datasets such as CIFAR-10, CIFAR-100, STL-10, Tiny ImageNet, and Oxford Flowers 102, as well as on a large-scale dataset, ImageNet. The detailed implementation settings are provided in Appendix A.

Small- and Medium-scale Datasets. Due to the challenges of binarizing DW convs, previous BNNs primarily relied on ResNet-18 or MobileNet V1 with regular convs replacing depth-wise operations. In contrast, our method successfully binarizes DW convs while maintaining competitive performance.

Table 2 shows the experimental results. BD-Net-A (ReActNet) reduces computational cost by approximately $3 \times$

Network	CIFAR-10		CIFAR-100		STL-10		Tiny ImageNet		Flowers 102	
	Acc	OPs	Acc	OPs	Acc	OPs	Acc	OPs	Acc	OPs
Full-Precision	92.26	42.6M	69.62	42.6M	65.97	383M	58.04	170M	50.31	569M
ReActNet	71.04	6.97M	57.19	6.97M	69.23	62.7M	42.62	27.9M	31.88	97M
BD-Net-A (ReActNet)	78.37	1.56M	50.11	1.56M	65.60	14.0M	38.98	6.23M	36.46	32.5M
BD-Net-B (ReActNet)	89.93	1.83M	63.83	1.83M	77.03	16.4M	52.06	7.31M	45.18	34.6M
AdaBin	81.60	7.28M	55.35	7.28M	69.68	65.6M	44.96	29.1M	40.66	102M
BD-Net-A (AdaBin)	84.87	1.59M	58.28	1.59M	65.19	14.3M	44.10	6.37M	41.16	33M
BD-Net-B (AdaBin)	90.48	1.86M	64.66	1.86M	73.90	16.7M	52.01	7.44M	45.70	35.5M

Table 2: Experimental Results for Different Networks on Various Datasets

Method	Backbone	Conv type	BOPs ($\times 10^9$)	FLOPs ($\times 10^8$)	OPs ($\times 10^8$)	Top-1 Acc. (%)	Top-5 Acc. (%)
BNN	ResNet	regular	1.70	1.20	1.47	42.2	67.1
XNOR-Net	ResNet	regular	1.68	1.40	1.66	51.2	73.2
Bi-real Net 18	ResNet	regular	1.68	1.41	1.67	56.4	79.5
XNOR-Net++	ResNet	regular	1.68	1.42	1.68	57.1	79.9
IR-NET	ResNet	regular	1.68	1.41	1.67	58.1	80.0
Real-to-Binary	ResNet	regular	1.68	1.40	1.66	65.4	86.2
ReCU	ResNet	regular	1.68	1.40	1.67	66.4	86.5
INSTA-BNN	ResNet	regular	1.68	1.43	1.70	67.6	87.5
ReActNet	MobileNet V1	regular	4.82	0.12	0.87	69.4	-
Adabin	MobileNet V1	regular	4.82	0.21	0.96	70.4	-
INSTA-BNN	MobileNet V1	regular	4.82	0.20	0.95	71.7	90.3
MoBiNet (K=4)	MobileNet V1	group	-	-	0.52	54.4	77.5
MoBiNet (K=3)	MobileNet V1	group	-	-	0.49	53.5	76.5
Binary MobileNet	MobileNet V1	group	-	-	0.33	51.1	74.2
*ReActNet 0.5 \times	MobileNet V1	regular	1.22	0.12	0.31	60.7	82.6
BD-Net-A (ReActNet)	MobileNet V1	depth-wise	1.09	0.13	0.30	63.9	84.8
BD-Net-B (ReActNet)	MobileNet V1	depth-wise	1.08	0.16	0.33	65.3	85.8
BD-Net-B(x1.5) (ReActNet)	MobileNet V1	depth-wise	2.50	0.26	0.65	69.4	88.8

Table 3: Comparison of various methods for different metrics. The reproduced result is marked with *.

to 4.5 \times compared to ReActNet while maintaining competitive performance. Similarly, BD-Net-A (AdaBin) improves accuracy by +3.27 p.p., +2.93 p.p., and +0.50 p.p. on CIFAR-10, CIFAR-100, and Oxford Flowers 102, respectively, while reducing OPs by 26%, 22%, and 34% on CIFAR datasets, Tiny ImageNet, and Oxford Flowers 102. Building on this, BD-Net-B (ReActNet) slightly increases OPs over BD-Net-A but consistently outperforms ReActNet across all datasets. Likewise, BD-Net-B (AdaBin) further improves accuracy over BD-Net-A with only a marginal increase in OPs. Specifically, BD-Net-B achieves accuracy gains of +4.22 p.p. to +9.31 p.p. on CIFAR-10, CIFAR-100, STL-10, Tiny ImageNet, and Oxford Flowers 102, while reducing OPs by a factor of 2.68 to 8.47.

Large-scale Dataset. Table 3 compares the performance of various methods on ImageNet, including ResNet-based methods, MobileNet V1-based methods with regular convs, and group conv-based methods.

ResNet-based methods achieve high accuracy but suffer from heavy computational costs. MobileNet V1-based meth-

ods, such as ReActNet, offer a better trade-off but still require more operations than group conv methods, which improve efficiency at the cost of reduced accuracy.

Our approach, BD-Net, addresses the limitations of previous methods by successfully incorporating DW convs into Binary Neural Networks. BD-Net-B achieves a Top-1 accuracy of 65.3% with only 33M OPs on ImageNet, reducing the number of operations by approximately three times compared to MobileNet V1 with regular convs.

To provide a fair comparison with approaches requiring similar computational resources, we evaluated ReActNet with half the number of channels. ReActNet 0.5 \times achieves a top-1 accuracy of 60.7% with 30M operations. In comparison, our method delivers over 4.6 p.p. higher performance with similar computational demands. This demonstrates that BD-Net not only reduces the required operations but also achieves superior performance through a more efficient architectural design.

Extension to Other Efficient Architectures. Due to the extreme reduction in representation capacity caused by bina-

ShuffleNet V1	CIFAR-10		CIFAR-100		STL-10		Tiny ImageNet		Flowers 102	
	Acc	OPs	Acc	OPs	Acc	OPs	Acc	OPs	Acc	OPs
Full-Precision	91.16	9.70M	67.16	9.70M	67.12	23.3M	49.88	10.3M	40.97	126M
ReActNet	17.33	1.32M	37.87	1.32M	59.41	2.97M	24.42	1.32M	37.99	16.2M
ReActNet w/ br	79.80	1.32M	51.77	1.32M	60.42	2.97M	40.10	1.32M	45.02	16.2M
BD-Net-B (ReAct w/ br)	81.57	0.87M	59.03	0.87M	64.31	1.96M	35.83	0.87M	44.10	10.7M

Table 4: Experimental results for Shufflenet V1 on various datasets. ‘br’ indicates the use of broadcast residual connections, where previous layer channels are broadcast to match the next layer’s channel dimensions for residual connections across layers with different channel counts.

MobileNet V3	CIFAR-10		CIFAR-100		STL-10		Tiny ImageNet		Flowers 102	
	Acc	OPs	Acc	OPs	Acc	OPs	Acc	OPs	Acc	OPs
Full-Precision	91.34	25.9M	72.29	25.9M	76.47	58.2M	59.76	25.9M	52.21	83.3M
ReActNet	19.08	9.80M	14.96	9.80M	24.08	6.26M	FAIL	9.80M	8.01	34.1M
ReActNet w/ br	86.21	9.80M	61.20	9.80M	67.24	6.26M	32.44	9.80M	49.64	34.1M
BD-Net-B (ReActNet w/ br)	85.92	1.14M	58.60	1.14M	69.22	1.47M	42.41	1.14M	51.33	7.99M

Table 5: Experimental results for MobileNet V3 on various datasets. ‘br’ indicates the use of broadcast residual connections, where previous layer channels are broadcast to match the next layer’s channel dimensions for residual connections across layers with different channel counts.

rization, existing BNNs have been applied only to ResNet and MobileNet V1. These networks have minimal variation in the number of channels across layers, enabling the existing layer-wise residual connections to be applied effectively. However, in ShuffleNet V1 and MobileNet V3, the channel dimensions vary continuously across layers, preventing the application of existing layer-wise residual connections and resulting in unstable training. To address this, we propose broadcast residual connections. This method constructs residual connections by replicating the channels of the previous layer to match the channel dimensions of the next layer, enabling extension to ShuffleNet V1 and MobileNet V3.

Tables 4 and 5 show the performance measurements of ReActNet and BD-Net-B on ShuffleNet V1 and MobileNet V3. Simply applying ReActNet resulted in highly unstable training and significant accuracy degradation. However, with the addition of broadcast residual connections, accuracy improvements ranged from 1.01 p.p. to 62.47 p.p. on ShuffleNet V1 and from 32.44 p.p. to 67.13 p.p. on MobileNet V3 across datasets. This is because layer-wise residual connections increase representation capacity, enabling the model to learn more diverse features. Nevertheless, applying ReActNet to ShuffleNet V1 and MobileNet V3 has limitations, as it reduces the OP count by only a factor of $2.4\times$ to $9.3\times$ by replacing 32-bit depth-wise convolutions with binary regular convolutions. In contrast, BD-Net-B achieves accuracy improvements ranging from 4.89 p.p. to 64.24 p.p. on ShuffleNet V1 and from 42.21 p.p. to 66.84 p.p. on MobileNet V3 while requiring significantly fewer operations than ReActNet. Further improvements are

Method	Accuracy (%)
Baseline	54.94
+ Pre-BN	56.93
+ 1.58-bit conv	56.18
Ours	58.28

Table 6: Ablation study of our proposed methods on MobileNet V1 with CIFAR-100 dataset.

required to effectively extend these methods beyond MobileNet V1 to the entire ShuffleNet and MobileNet series.

Ablation Study

To validate the effectiveness of each component in our proposed BD-Net architecture, we conduct an ablation study using MobileNet V1 on the CIFAR-100 dataset. As shown in Table 6, we use a naive binary DW conv as the baseline, achieving 54.94% accuracy. Introducing the Pre-BN residual connection improves accuracy to 56.93% yielding a +1.99 p.p. gain. This result supports our theoretical analysis, where the Pre-BN residual structure stabilizes optimization by improving the Hessian condition number. Replacing the naive binary DW conv with the proposed 1.58-bit conv increases accuracy to 56.18%, a +1.24 p.p. improvement over the baseline. This demonstrates that slightly increasing bit-width enhances the representational capacity of DW convs while maintaining a low computational cost.

Finally, combining both techniques in BD-Net achieves

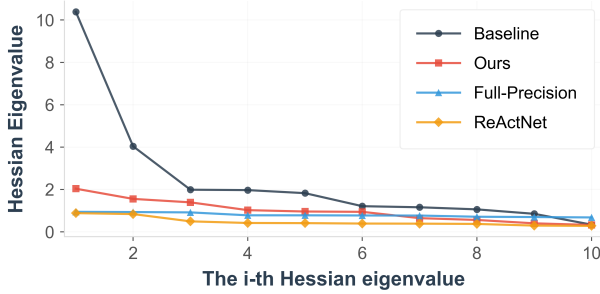


Figure 5: Graph of the top-10 hessian eigenvalues for several networks.

the highest accuracy of 58.28%, with a +3.34 p.p. improvement over the baseline. These results demonstrate the complementary effects of the Pre-BN residual connection, which stabilizes optimization, and the 1.58-bit conv, which enhances feature representation. More extensive ablation studies on multiple convs are provided in Appendix B.

Analysis

Hessian Eigenvalue. Figure 5 illustrates the top-10 Hessian eigenvalues of the MobileNet V1 backbone across various methods: Baseline (simply binarized MobileNet V1), ReActNet, Full-precision, and our proposed approach. The graph shows that the baseline method, in comparison to the full-precision network, presents a notably higher maximum eigenvalue, leading to significant training instability. Our proposed method effectively reduces the maximum Hessian eigenvalue, thereby lowering the condition number of the Hessian and improving training stability. Moreover, our BD-Net exhibits an eigenvalue distribution similar to ReActNet with more parameters than ours, and to MobileNet in full precision. This indicates that our method is highly effective in improving learning efficiency in environments with fewer parameters and lower quantization bits.

Loss Landscape. To assess the effect of our proposed method on training stability, we visualize the loss landscape using a random filter-normalized direction in Figure 6.

Comparing with the binarized MobileNet V1 shown in Figure 6a, the landscape with 1.58-bit conv Figure 6b appears significantly smoother. This demonstrates how the representational ability of a binarized model impacts training. Our 1.58-bit conv approach enhances the model’s representational ability in a parameter-efficient manner, and increases training stability.

Figure 6c demonstrates how effectively the loss landscape can be smoothed by the pre-BN residual which effectively reduces the condition number. It slightly increases the computational cost without model parameter increase, but it has a significant effect on smoothing the loss landscape. It is an effective structural compensation that mitigates the challenge originated from the structural limitations of DW convs. BD-Net, which incorporates both 1.58-bit conv and pre-BN residuals, demonstrates a much smoother loss landscape, enhancing both training stability and gen-

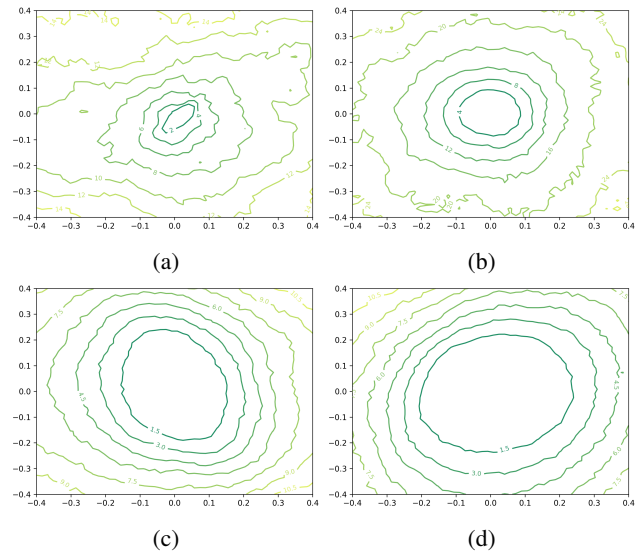


Figure 6: Loss landscapes on the CIFAR-100 dataset: (a) Binarized MobileNet V1 (same as Figure 2a), (b) Binarized MobileNet V1 with 1.58-bit conv, (c) Binarized MobileNet V1 with pre-BN residual, and (d) Our proposed method.

eralization capabilities. The landscape of BD-Net shows a level of smoothness very similar to that of full-precision depicted in Figure 2b. We further analyze the 3D loss landscapes of the baseline and BD-Net, revealing that BD-Net exhibits a smoother and more stable surface during optimization (see Appendix B).

Conclusion

We addressed the fundamental challenge of integrating binary DW convs into BNNs through comprehensive analysis. Our BD-Net introduces pre-BN residual connections and 1.58-bit convs, enabling the first successful binarization of DW convs in BNNs. Our method achieves the lowest computational cost with only 33M OPs on ImageNet while delivering up to 9.3 p.p. accuracy improvements across multiple datasets on MobileNet V1, demonstrating the potential for truly efficient binary neural networks. Furthermore, we extended our approach to ShuffleNet V1 and MobileNet V3, demonstrating its adaptability to other network architectures.

Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT)(RS-2019-II190421, Artificial Intelligence Graduate School Program (Sungkyunkwan University), 20%), (No.1711195788, Development of Flexible SW/HW Conjunctive Solution for on-edge self-supervised learning, 20%), (No.RS-2025-25442569, AI Star Fellowship Support Program (Sungkyunkwan Univ.), 20%), National Research Foundation of Korea (NRF) grants funded by the Korea government (MEST)(RS-2024-00352717, 20%) and by RCN-NRF

project AURoRA (RCN no. 359216, NRF grant RS-2025-03522980, 20%).

References

- Bulat, A.; and Tzimiropoulos, G. 2019. Xnor-net++: Improved binary neural networks. *arXiv preprint arXiv:1909.13863*.
- Coates, A.; Ng, A.; and Lee, H. 2011. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 215–223. JMLR Workshop and Conference Proceedings.
- Finkelstein, A.; Almog, U.; and Grobman, M. 2019. Fighting quantization bias with bias. *arXiv preprint arXiv:1906.03193*.
- Gluska, S.; and Grobman, M. 2020. Exploring neural networks quantization via layer-wise quantization analysis. *arXiv preprint arXiv:2012.08420*.
- Gou, J.; Yu, B.; Maybank, S. J.; and Tao, D. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6): 1789–1819.
- Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; and Xu, C. 2020. Ghostnet: More features from cheap operations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1580–1589.
- Han, S.; Pool, J.; Tran, J.; and Dally, W. 2015. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28.
- He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.
- Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.
- Howard, A.; Sandler, M.; Chu, G.; Chen, L.-C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. 2019. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision*, 1314–1324.
- Howard, A. G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; and Adam, H. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.
- Jacob, B.; Kligys, S.; Chen, B.; Zhu, M.; Tang, M.; Howard, A.; Adam, H.; and Kalenichenko, D. 2018. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2704–2713.
- Kim, N.-r.; Lee, J.-S.; and Lee, J.-H. 2024. Learning with Structural Labels for Learning with Noisy Labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 27610–27620.
- Krizhevsky, A.; and Hinton, G. 2009. Learning Multiple Layers of Features from Tiny Images. Technical Report UTML TR 2009-003, University of Toronto, Toronto, ON, Canada.
- Lee, C.; Kim, H.; Park, E.; and Kim, J.-J. 2023. INSTA-BNN: Binary neural network with instance-aware threshold. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 17325–17334.
- Liang, T.; Glossner, J.; Wang, L.; Shi, S.; and Zhang, X. 2021. Pruning and quantization for deep neural network acceleration: A survey. *Neurocomputing*, 461: 370–403.
- Lin, X.; Zhao, C.; and Pan, W. 2017. Towards accurate binary convolutional neural network. *Advances in neural information processing systems*, 30.
- Liu, Z.; Shen, Z.; Savvides, M.; and Cheng, K.-T. 2020. Reactnet: Towards precise binary neural network with generalized activation functions. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, 143–159. Springer.
- Liu, Z.; Wu, B.; Luo, W.; Yang, X.; Liu, W.; and Cheng, K.-T. 2018. Bi-real net: Enhancing the performance of 1-bit cnns with improved representational capability and advanced training algorithm. In *Proceedings of the European conference on computer vision (ECCV)*, 722–737.
- Long, J.; Shelhamer, E.; and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3431–3440.
- Ma, R.; Qiao, G.; Liu, Y.; Meng, L.; Ning, N.; Liu, Y.; and Hu, S. 2024. A&B BNN: Add&Bit-Operation-Only Hardware-Friendly Binary Neural Network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5704–5713.
- Marteau-Ferey, U.; Bach, F.; and Rudi, A. 2019. Globally convergent newton methods for ill-conditioned generalized self-concordant losses. *Advances in Neural Information Processing Systems*, 32.
- Minaee, S.; Boykov, Y.; Porikli, F.; Plaza, A.; Kehtarnavaz, N.; and Terzopoulos, D. 2021. Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(7): 3523–3542.
- Nagel, M.; Fournarakis, M.; Bondarenko, Y.; and Blankevoort, T. 2022. Overcoming oscillations in quantization-aware training. In *International Conference on Machine Learning*, 16318–16330. PMLR.
- Nilsback, M.-E.; and Zisserman, A. 2008. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, 722–729. IEEE.
- Phan, H.; He, Y.; Savvides, M.; Shen, Z.; et al. 2020a. Mobinet: A mobile binary network for image classification. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 3453–3462.
- Phan, H.; Liu, Z.; Huynh, D.; Savvides, M.; Cheng, K.-T.; and Shen, Z. 2020b. Binarizing mobilenet via evolution-based searching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 13420–13429.
- Rakka, M.; Fouda, M. E.; Khargonekar, P.; and Kurdahi, F. 2022. Mixed-precision neural networks: A survey. *arXiv preprint arXiv:2208.06064*.

Rastegari, M.; Ordonez, V.; Redmon, J.; and Farhadi, A. 2016. Xnor-net: Imagenet classification using binary convolutional neural networks. In *European conference on computer vision*, 525–542. Springer.

Rawat, W.; and Wang, Z. 2017. Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9): 2352–2449.

Redmon, J.; Divvala, S.; Girshick, R.; and Farhadi, A. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 779–788.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115: 211–252.

Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; and Chen, L.-C. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4510–4520.

Sheng, T.; Feng, C.; Zhuo, S.; Zhang, X.; Shen, L.; and Aleksic, M. 2018. A quantization-friendly separable convolution for mobilenets. In *2018 1st Workshop on Energy Efficient Machine Learning and Cognitive Computing for Embedded Applications (EMC2)*, 14–18. IEEE.

Shin, J.; So, J.; Park, S.; Kang, S.; Yoo, S.; and Park, E. 2023. NIPQ: Noise proxy-based integrated pseudo-quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3852–3861.

Tu, Z.; Chen, X.; Ren, P.; and Wang, Y. 2022. Adabin: Improving binary neural networks with adaptive binary sets. In *European conference on computer vision*, 379–395. Springer.

Wang, K.; Liu, Z.; Lin, Y.; Lin, J.; and Han, S. 2019. Haq: Hardware-aware automated quantization with mixed precision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8612–8620.

Yang, X. 2015. Tiny imagenet visual recognition challenge.

Zhang, X.; Zhou, X.; Lin, M.; and Sun, J. 2018. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6848–6856.

Zhou, S.; Wu, Y.; Ni, Z.; Zhou, X.; Wen, H.; and Zou, Y. 2016. Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients. *arXiv preprint arXiv:1606.06160*.

Zou, Z.; Chen, K.; Shi, Z.; Guo, Y.; and Ye, J. 2023. Object detection in 20 years: A survey. *Proceedings of the IEEE*, 111(3): 257–276.