AB-STE: Adaptive Blended Gradient Estimation for Efficient Binarized Networks

Siddharth Gupta (✉) and Akash Kumar

Ruhr-Universität Bochum, Germany
{siddharth.gupta, akash.kumar}@rub.de

Abstract. Binary Neural Networks (BNNs) offer a highly efficient alternative to traditional deep learning models by drastically reducing memory and computational demands, making them well-suited for deployment in resource-constrained environments like edge devices. Despite their efficiency, BNNs are often limited by inaccurate and unstable gradient estimation using traditional Straight Through Estimator (STE) methods, which disrupt gradient flow and impede convergence. BinaryConnect introduced STE to approximate the gradients of the sign function; however, this approximation causes significant inconsistencies, ultimately compromising training stability. While various methods have been proposed to address these issues, many fail to consider that minimizing estimation error can inadvertently reduce gradient stability. Such highly divergent gradients can increase the risk of vanishing or exploding gradients, thereby hindering effective training. In this paper, we propose two novel Adaptive Blended Straight Through Estimators (*AB-STE*): *AB-ArcTan-STE* and *AB-Tanh-STE*. Unlike previous methods, AB-STE blends a linear component with a non-linear function to provide both stability and expressiveness during training, addressing key challenges faced by BNNs. By combining the simplicity of linearity with the representational power of non-linearity, AB-STE maintains a balanced gradient flow throughout training, ensuring both stability and effective learning. Extensive experiments on CIFAR-10 and ImageNet demonstrate that AB-STE achieves superior performance, surpassing existing state-of-the-art methods. Specifically, our *AB-Tanh-STE* achieved an accuracy of 94.60% on ResNet-18 for CIFAR-10, and a Top-1 accuracy of 67.96% on ImageNet, demonstrating the effectiveness of our adaptive blending strategy in enhancing training stability and accuracy. Notably, the parameters were binarized to achieve efficiency while maintaining stable gradient flow.

Keywords: Binary Neural Networks (BNNs) · Straight Through Estimator (STE) · Gradient Estimation · Quantization · Deep Learning.

1 Introduction

Deep neural networks (DNNs), particularly convolutional neural networks (CNNs), have achieved remarkable success across a wide range of computer vision tasks,

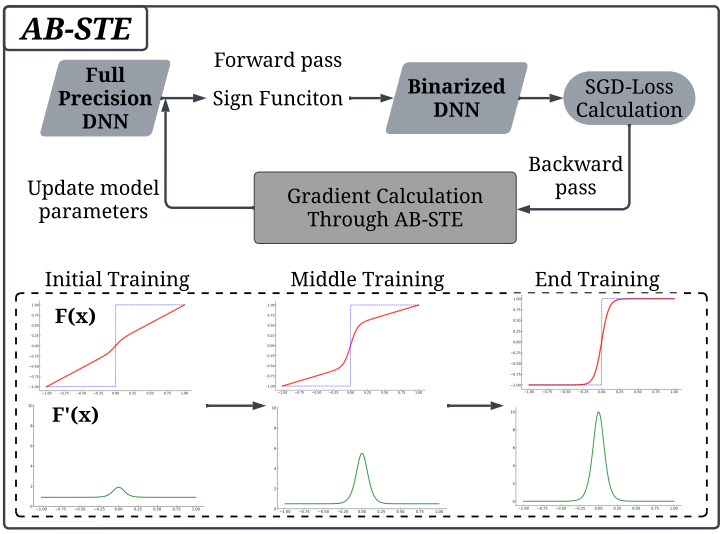


Fig. 1: Proposed framework for *AB-STE*

such as image classification [2, 3], object detection [4, 5], and semantic segmentation [6]. Despite their success, the deployment of these models on resource-constrained edge devices, like mobile phones, smartwatches, and cameras, is challenging due to their large number of parameters and high computational demands. To address these challenges, binarizing DNNs has emerged as a promising approach, providing a significant reduction in memory footprint and computational costs [7]. Binarizing the parameters of DNNs makes them easier to deploy on hardware since convolution operations can be implemented as efficient bit-wise operations [9]. However, a major challenge in binarization is the inability to propagate gradients effectively through binary activations, which leads to poor model accuracy and hinders the training of deep architectures [10, 11].

BinaryConnect [7] and BinaryNet [8] were among the first approaches to binarize both weights and activations. These methods employed the straight-through estimator (STE) to approximate the gradients of the sign function during back-propagation, which led to significant improvements in training binarized networks. However, the inconsistency between the forward pass (binarizing weights and activations) and the backward pass (approximating gradients) introduces a critical problem, leading to poor gradient flow and reduced accuracy in deeper networks. To mitigate the issues caused by traditional STE, ReSTE [1] proposed a balanced approach to stabilize gradients similar to STE, while incorporating flexibility through a power function. However, ReSTE's effectiveness is limited near zero values, resulting in gradients that are not sufficiently smooth. Although previous methods attempted to narrow the estimation error, they often led to

divergent gradients. This motivated our design of *AB-STE*, which introduces an adaptive approach to ensure both stability and effective gradient flow.

The proposed *Adaptive Blended Straight Through Estimator (AB-STE)* employs a blend of two components: a linear component for stability and a non-linear component for expressiveness, which approximates the sign function. This adaptive blend evolves throughout training to maintain gradient stability while enhancing the model's representational capacity. In the initial phase, the forward function, $F(x)$, behaves similarly to a linear function ($y = x$), providing stability with low gradient magnitudes for the backward function, $F'(x)$. As training progresses, $F(x)$ evolves to approximate the non-linear characteristics of the sign function, resulting in higher gradient magnitudes through $F'(x)$. This progression enhances the model's ability to learn more complex representations while maintaining a stable gradient flow. Fig. 1 illustrates the training framework of *AB-STE*, highlighting how the adaptive evolution of $F(x)$ and $F'(x)$ helps improve both stability and learning during different training stages. Our function moves from a simple linear approximation to a more step-like behavior, while maintaining smooth gradients throughout backpropagation to support effective training. We further illustrate the forward and backward passes of the proposed estimator in Fig. 2. The plots demonstrate how our function transitions from an initial STE-like behavior to a more refined step-function approximation over the course of training while maintaining smooth gradients throughout. This ensures both effective approximation and stable gradient flow, facilitating robust and efficient model training.

The main contributions of this paper are as follows:

Adaptive Blended Straight-Through Estimators (AB-STE): We propose two novel adaptive blended estimators, *AB-Tanh-STE* and *AB-ArcTan-STE*, which combine linear and non-linear components to stabilize and improve gradient flow during BNN training.

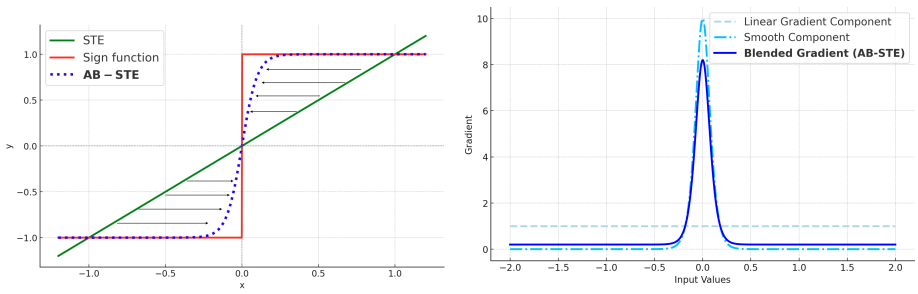
Enhanced Gradient Stability and Smoothness Our approach addresses the inconsistency problem in traditional STE methods by balancing gradient smoothness and stability, reducing the risk of vanishing or exploding gradients.

Blended Function for Forward and Backward Passes The proposed blended function incorporates a linear component for stability and a non-linear component for expressiveness, enabling the transition from STE-like behavior to step-function approximation during training.

Extensive Experimental Validation Experiments conducted on CIFAR-10 and ImageNet datasets demonstrate the superior performance of *AB-STE* compared to existing state-of-the-art estimators for BNNs.

Open-Source Implementation We provide an open-source implementation of *AB-STE* (`https://github.com/sid-3dev/AB_STE`) to encourage reproducible research and further development in efficient neural network training.

The remainder of this paper is organized as follows: Section 2 reviews the state-of-the-art techniques for DNN binarization. Section 3 introduces our proposed methods and their theoretical analysis, followed by the results and analysis in Section 4. Finally, Section 5 provides concluding remarks and insights.



(a) Forward Pass (b) Backward Pass

Fig. 2: An intuitive illustration of adaptive blending of linear smooth components to show how our function moves from STE to Sign function approximation while keeping the gradient smooth during backpropagation for improved BNN training.

2 Related Work

Network binarization seeks to enhance the speed of neural network inference while significantly reducing memory requirements, all with minimal accuracy loss. One effective strategy to achieve this is by employing bitwise operations in low-precision networks. By converting 32-bit parameters such as weights and activations into binary form, computational efficiency is considerably boosted, and memory consumption is greatly decreased. BinaryConnect [7] and BinaryNet [8] were pioneering approaches that focused on binarizing network weights and both weights and activations, respectively, for use during both training and inference. These works utilized straight-through estimators to enable training of deep neural networks with binarized parameters, particularly addressing the non-differentiability issue that arises during the binarization process. In binarized neural networks, weights and activations are often represented using a sign function, which complicates gradient computation through standard backpropagation due to its discontinuous nature. To circumvent this problem, STE approximates the backward gradient, allowing effective network training. During the forward pass, a binarization function such as the sign function is used:

$$x_b = \text{sign}(x) = \begin{cases} +1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases} \tag{1}$$

However, the gradient of this function is zero almost everywhere, making it unsuitable for backpropagation. In STE, the gradient is approximated in a simpler form, treating the forward binarization as an identity function during the backward pass. This can be mathematically represented as:

$$\frac{\partial L}{\partial x} \approx \frac{\partial L}{\partial x_b} \cdot \mathbf{1}_{|x| \leq 1} \tag{2}$$

where L is the loss function, x_b is the binarized version of x, and $\mathbf{1}_{|x|\leq 1}$ is an indicator function that constrains the gradient to pass through values of x in the range $[-1, 1]$. This simple approximation effectively allows gradients to flow through the network, enabling the model to learn and adapt weights despite the non-differentiable nature of the binarization step. Using STE, the model can be trained to achieve comparable accuracy to its full-precision counterpart, while greatly benefiting from reduced complexity and memory footprint.

Despite the use of STE, the accuracy of binary networks remains significantly lower compared to their full-precision counterparts. Various strategies have been proposed to address this issue. [12] introduced architectural changes to enhance the expressiveness of binary networks, though these improvements depend heavily on modifying network architecture. Other methods, such as knowledge distillation and additional regularizations [13, 14], aim to improve training but often result in increased computational costs during training.

Many studies have focused on improving gradient estimation in binary neural networks (BNNs). For instance, Bi-Real-Net [12] employs a piece-wise polynomial, DSQ [15] introduces a tanh-based function, IR-Net uses an error decay estimator function, and FDA [16] applies Fourier series to improve gradient computation. While these methods have demonstrated good performance, they often overlook the importance of gradient stability. Reducing estimation error too aggressively can lead to highly divergent gradients, increasing the risk of gradient vanishing or exploding, which ultimately impairs effective training.

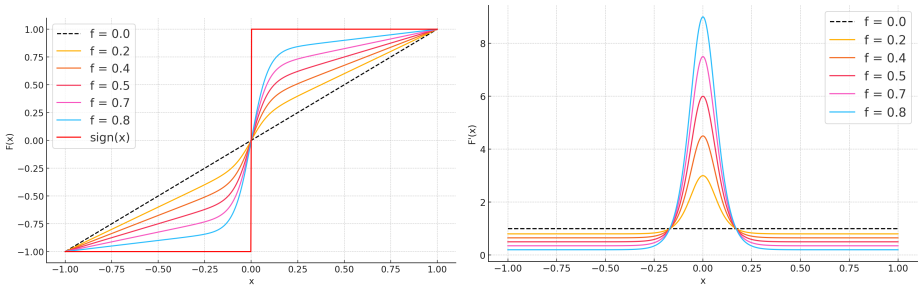
The authors of ReSTE [1] proposed a method for stable gradient calculation; however, the resulting gradient space is not smooth, which limits training capability during binarization. To address these challenges, we propose an **Adaptive Blended Straight-Through Estimator (AB-STE)**. Compared to other estimators, our approach provides stable training with smooth gradients, resulting in better overall performance. Extensive experiments show that our method surpasses existing state-of-the-art methods, effectively addressing both the gradient stability and smoothness challenges in binarized networks.

3 Proposed Techniques

3.1 Adaptive Blended Straight Through Estimator (AB-STE)

The authors of ReSTE [1] demonstrate that the sign function and STE represent two extremes in terms of gradient stability. The sign function has zero gradients almost everywhere and an infinite gradient at the origin, leading to either vanishing or exploding gradients, resulting in high gradient instability. In contrast, STE approximates the gradients of the sign function using a linear function, which does not alter the backward gradient during estimation.

Considering these characteristics, we designed an estimator for gradients that balances stability and expressiveness. Therefore, we introduce a blend of linearity and non-linearity in the estimator that aims to reduce the discrepancy between forward and backward computations. Eq. (3) represents the forward pass for the



(a) Forward Pass (b) Backward Pass

Fig. 3: Showing the behaviour of Advaptive Blended Tanh based Straight Through Estimator (AB-Tan-STE) with different factor (f) values

proposed AB-STE. By incorporating a blending factor f and a tunable scaling parameter k, the model can adaptively adjust the ratio between the linear and tanh components based on the training dynamics, allowing a more flexible and accurate representation of the gradients.

$$F(x) = (1 - f) \cdot x + f \cdot \tanh(kx) \tag{3}$$

where f controls the blend between the linear and non-linear parts, and k is a scaling factor for x. When $f = 0$, this blending function behaves as a standard STE, and when $f = 1$, it functions similarly to the tanh function (as an approximation of the sign function). During training, we start with low values of f and gradually increase it as the number of epochs progresses. This approach allows us to initially leverage the stability of STE and gradually incorporate the expressiveness of the tanh function. In Fig. 3, we illustrate the changes in the function as f changes. It can be observed that $F(x)$ behaves more like an STE when f is small and more like a sign function as f increases.

backward pass function $F'(x)$ is given by:

$$F'(x) = (1 - f) + f \cdot k \cdot \operatorname{sech}^2(kx) \tag{4}$$

Fig. 3 also shows the derivative $F'(x)$. As f increases, $F'(x)$ provides a high and smooth peak near zero, indicating the function's ability to effectively calculate gradients throughout training, thereby ensuring gradient stability and reducing the risk of gradient vanishing or exploding. Furthermore, we introduce another adaptive blended function for straight-through estimation called **AB-ArcTan-STE**. This estimator blends a linear function with an arctan function, as defined in Eq. (5). The use of the arctan function in this blend aims to extend the range of the resulting function, which allows the model to retain gradient information over a broader spectrum of input values. This makes the estimator more effective, particularly when handling larger values of x. The arctan function also provides a more gradual gradient decay, contributing to increased stability

Algorithm 1 Adaptive Blended Straight-Through Estimator (AB-STE)

Input: Training dataset: X, epochs: N, learning rate: η, initial blending factor: f_0, scaling parameter: k
Initialize model parameters θ
for*epoch* $= 1 \ldots$ N
for*batch* $B \subset X$
Forward Pass: Compute activations using the sign function:

$$y \leftarrow \text{sign}(x)$$

Calculate loss $L(\theta)$ using output y
Backward Pass: Compute gradient of the loss with respect to parameters using the blended gradient function:

$$F'(x) \leftarrow (1-f) + f \cdot k \cdot \text{sech}^2(kx) \quad \text{(for AB-Tanh-STE)}$$
$$F'(x) \leftarrow \frac{1-f}{2} + \frac{f \cdot k}{\arctan(2k) \cdot (1 + (k \cdot x)^2)} \quad \text{(for AB-ArcTan-STE)}$$

Use $F'(x)$ to compute parameter gradients (not applied to activations)
Update model parameters using stochastic gradient descent:

$$\theta \leftarrow \theta - \eta \cdot \nabla_\theta L(\theta)$$

Update blending factor f (e.g., linearly or exponentially increase f with epochs)

and smoother gradient flow during training. The backward pass of AB-ArcTan-STE is presented in Eq. (6).

$$F(x) = \frac{1-f}{2} \cdot x + \frac{f}{\arctan(2k)} \cdot \arctan(k \cdot x) \tag{5}$$

$$F'(x) = \frac{1-f}{2} + \frac{f \cdot k}{\arctan(2k) \cdot (1 + (k \cdot x)^2)} \tag{6}$$

$\arctan(x)$ is the angle between $-\frac{\pi}{2}$ and $\frac{\pi}{2}$ radians whose tangent is x.

Further, we present an Algorithm 1 that utilizes the Adaptive Blended Straight-Through Estimator (AB-STE) for training binarized models. In the forward pass, activations are computed using the sign function to maintain binarized representation. During the backward pass, a blended gradient function $F'(x)$, combining linear and non-linear components, is used to compute parameter gradients, enabling smooth and adaptive gradient updates. The blending factor f is gradually increased throughout training to transition from stable linear gradients to more expressive non-linear gradients, effectively balancing stability and learning capacity. This approach helps mitigate gradient instability, ensuring efficient training of binarized models.

3.2 Theoretical Analysis of Adaptive Blended Straight-Through Estimator (AB-STE)

We provide a theoretical analysis of the proposed Adaptive Blended Straight Through Estimator, focusing on stability, convergence, and gradient properties. Specifically, we analyze the impact of the blending factor f and scaling parameter k on gradient flow, stability, and convergence.

Gradient Stability and Variance Lemma 1 (Gradient Stability): The variance of the gradient $F'(x)$ decreases as the blending factor f increases from 0 to 1, resulting in a smoother gradient update.

Proof:

1. Consider the derivative function $F'(x)$. The variance $\mathrm{Var}(F'(x))$ depends on the value of f and the distribution of the input x.
2. When $f = 0$, $F'(x) = 1$, which results in no variance in gradient values.
3. When $f > 0$, the variance of $F'(x)$ depends on the contribution of the term $f \cdot k \cdot \mathrm{sech}^2(kx)$.
4. Since $\mathrm{sech}^2(kx)$ is bounded between 0 and 1, the variance of the gradient is limited and depends on f and k. As f increases, the contribution of the non-linear term becomes more prominent, resulting in a more stable and smoothed gradient.

Implication: This analysis indicates that increasing the blending factor f results in a more stable gradient update, which is crucial for reducing the risk of sudden changes in gradient values, thus improving overall training stability.

Smoothness of Gradient Flow The smoothness of the gradient is crucial to avoid vanishing or exploding gradients during training. The smoothness property of the proposed estimator is analyzed using the second-order derivative of $F(x)$.

Lemma 2 (Smoothness of Gradient Flow): The gradient $F'(x)$ of AB-STE is Lipschitz continuous with a Lipschitz constant that depends on the blending factor f and the scaling parameter k.

Proof:

1. The second derivative of $F(x)$ is given by:
$$F''(x) = f \cdot k^2 \cdot \mathrm{sech}^2(kx) \cdot \tanh(kx)$$
2. The Lipschitz constant L for $F'(x)$ can be bounded by the maximum value of $|F''(x)|$:
$$L \leq f \cdot k^2$$
3. Since $f \in [0, 1]$ and $\mathrm{sech}^2(kx) \leq 1$, the Lipschitz constant depends linearly on f and quadratically on k. This implies that the smoothness of the gradient increases with smaller values of k, while larger values of k can result in sharper changes in gradient, potentially causing instability.

Implication: Ensuring that the Lipschitz constant is appropriately controlled helps maintain smooth gradient flow, thereby reducing the risk of gradient explosion or vanishing, particularly in deep networks.

Gradient Flow Improvement Furthermore, when $f \to 1$, the gradient $F'(x)$ has a peak near zero that helps maintain sufficient gradient flow through the layers, especially during backpropagation. This prevents gradients from vanishing in deeper layers and improves convergence.

In summary, the theoretical analysis shows that AB-STE effectively maintains gradient stability and smoothness through careful control of the blending factor f and the scaling parameter k. The variance of the gradient $F'(x)$ is reduced as f increases, ensuring stable updates, while the Lipschitz continuity of $F'(x)$ guarantees smooth gradient flow, reducing the risk of vanishing or exploding gradients. By gradually increasing f, AB-STE ensures effective gradient flow throughout the network, leading to improved convergence during training, particularly for deep binarized models.

4 Experimental Setup and Results

4.1 Datasets and Training Setup

This study uses two popular datasets commonly employed in the binary neural network literature: **CIFAR-10** [18] and **ImageNet ILSVRC-2012** [17].

The CIFAR-10 dataset consists of 50,000 training images and 10,000 testing images across 10 categories, with each image having a resolution of 32×32 and three RGB color channels. The ImageNet ILSVRC-2012 dataset is a large-scale dataset with over 1.2 million training images and 50,000 validation images, each at a resolution of 224×224, covering 1,000 categories.

To ensure a fair comparison with existing methods, we adopted similar training settings as other binary methods [1,19,20]. We used pre-processing techniques such as RandomCrop, RandomHorizontalFlip, and Normalize for both CIFAR-10 and ImageNet. The models were trained using Stochastic Gradient Descent (SGD) with an initial learning rate of 0.1, and a cosine learning rate decay schedule was employed to gradually reduce the learning rate during training.

For CIFAR-10, the models were trained for 1,000 epochs, while for ImageNet, training was conducted for 250 epochs. The hyperparameter k, which controls the iterative nature of the adaptive blending, was fixed at 10 throughout the experiments. We varied the blending factor f between 0.2 and 0.8 to study its impact on training performance and gradient stability. Importantly, parameter quantization was employed during training, whereas activations were kept at full precision to maintain expressive feature representations and ensure stable gradient flow.

4.2 Results and Analysis

The proposed Adaptive Blended Straight-Through Estimators (AB-STE) were evaluated using the CIFAR-10 and ImageNet datasets, focusing on both accuracy and training stability compared to the existing state-of-the-art methods. In these experiments, we applied parameter binarization while retaining full-precision activations, which allowed us to effectively maintain gradient stability and minimize the impact of quantization on the training dynamics.

Results on CIFAR-10 Table 1 summarizes the results of the CIFAR-10 experiments across different architectures, including VGG-small, ResNet-18, and ResNet-20. We compare our proposed AB-Tanh-STE and AB-ArcTan-STE with other established methods like DoReFa-Net [20], IR-Net [19], and ReSTE [1]. Our AB-Tanh-STE and AB-ArcTan-STE consistently demonstrated performance that was comparable to or better than the current state-of-the-art methods.

The AB-Tanh-STE method, in particular, achieved the highest accuracy across all three architectures, with an accuracy of **93.16%** on VGG-small and **94.60%** on ResNet-18, which were very close to their floating-point counterparts. This highlights the effectiveness of our adaptive blending strategy in maintaining stability while achieving high accuracy, despite parameter binarization.

Table 1: Accuracy Comparison on CIFAR-10 Across Different Architectures. The proposed AB-Tanh-STE and AB-ArcTan-STE methods consistently achieve competitive accuracy compared to floating-point and state-of-the-art binary methods, using parameter quantization while keeping activations at full precision.

Architecture	**Method**	**Accuracy (%)**
VGG-small	Floating Point	93.30
	DoReFa-Net [20]	92.13
	IR-Net [19]	90.92
	ReSTE [1]	92.53
	AB-Tanh-STE (ours)	**93.16**
	AB-ArcTan-STE (ours)	93.00
ResNet-18	Floating Point	94.86
	DoReFa-Net [20]	94.13
	IR-Net [19]	94.33
	ReSTE [1]	93.68
	AB-Tanh-STE (ours)	**94.60**
	AB-ArcTan-STE (ours)	94.18
ResNet-20	Floating Point	91.74
	DoReFa-Net [20]	90.79
	IR-Net [19]	91.03
	ReSTE [1]	91.32
	AB-Tanh-STE (ours)	**91.54**
	AB-ArcTan-STE (ours)	91.12

Results on ImageNet The results on the ImageNet dataset are presented in Table 2. For ResNet-18, our methods show clear improvements in Top-1 and Top-5 accuracy compared to previous estimators. Specifically, AB-Tanh-STE achieved a **Top-1 accuracy of 67.96%** and AB-ArcTan-STE achieved a **Top-5 accuracy of 87.66%**, surpassing the ReSTE baseline. For ResNet-34, AB-

Tanh-STE achieved a **Top-1 accuracy of 71.31%** and a **Top-5 accuracy of 89.98%**, which outperformed ReSTE and demonstrated the effectiveness of our proposed method on larger architectures.

The adaptive blending strategy enables our models to maintain a stable training process, even on a challenging dataset such as ImageNet, where the vast number of categories and high resolution of images pose significant challenges for binary neural networks. By using parameter quantization with full-precision activations, our methods demonstrate that adaptive blending not only stabilizes training but also provides an effective means to achieve high accuracy.

Table 2: Accuracy Comparison on ImageNet ILSVRC-2012 Across Different Architectures. The proposed AB-Tanh-STE and AB-ArcTan-STE methods show improvements in Top-1 and Top-5 accuracy while using parameter quantization and full-precision activations, highlighting their effectiveness on large-scale datasets.

Architecture	**Method**	**Top-1 Acc (%)**	**Top-5 Acc (%)**
ResNet-18	Floating Point	69.58	89.19
	ReSTE [1]	67.66	87.48
	AB-Tanh-STE (ours)	**67.96**	87.64
	AB-ArcTan-STE (ours)	67.68	**87.66**
ResNet-34	Floating Point	73.32	91.27
	ReSTE [1]	70.66	89.43
	AB-Tanh-STE (ours)	**71.31**	**89.98**
	AB-ArcTan-STE (ours)	71.18	89.71

4.3 Effect of Blending Factor

During training, the blending factor f was linearly increased from **0.2** to **0.8**. This approach allowed the model to benefit from stable training in the initial epochs while gradually introducing non-linearity to enhance representational capacity. Our results indicate that such adaptive blending, combined with parameter quantization and full-precision activations, is crucial for achieving an optimal balance between gradient smoothness and model expressiveness, leading to consistent improvements in both CIFAR-10 and ImageNet benchmarks.

The experiments on CIFAR-10 and ImageNet demonstrate that the proposed **AB-STE** methods outperform traditional binary neural network training techniques by effectively balancing gradient stability and expressiveness. The use of a blended function allows us to avoid the gradient vanishing and exploding issues that often hinder STE-based training methods.

4.4 Computational Resources Analysis

Table 3: Comparison of FLOPs and Training Time Across Methods

Method	FLOPs per Input	CIFAR-10 Time (s/epoch)	ImageNet Time (mm:ss/epoch)	CIFAR-10 Acc. (%)	ImageNet Acc. (%)
ReSTE	10 (5 for power, 4 for comparisons, 1 for sign)	16	10:56	93.68	67.66
AB-Tanh-STE	6 (2 for tanh, 4 for blending)	14	10:33	94.60	67.96
AB-ArcTan-STE	7 (3 for arctan, 4 for blending)	12	10:22	94.18	67.68

To evaluate computational efficiency, we conducted a FLOP (floating point operations) analysis for backpropagation and measured the training time per epoch for each method. These analyses help assess the computational cost of the proposed approaches in comparison to ReSTE. Table 3 summarizes the FLOP requirements per input element during backpropagation, alongside training time per epoch on CIFAR-10 and ImageNet using the ResNet-18 architecture. As observed, AB-Tanh-STE and AB-ArcTan-STE require fewer FLOPs per input compared to ReSTE, resulting in notable speedups in training time. The reduction in FLOPs is achieved by eliminating power operations and reducing conditional checks, thereby improving computational efficiency.

ReSTE's backpropagation operation includes multiple conditional checks and power operations, leading to higher computational cost and longer training times. In contrast, AB-Tanh-STE and AB-ArcTan-STE leverage simpler mathematical functions such as tanh and arctan, which require fewer computations. Optimized gradient flow reduces unnecessary computations while preserving accuracy. Efficient FLOP reduction enables 14% and 25% speedups on CIFAR-10 and 4% and 6% speedups on ImageNet, respectively. These results highlight that AB-Tanh-STE and AB-ArcTan-STE not only improve accuracy but also significantly reduce computational costs, making them more efficient for large-scale BNN training. To balance hardware efficiency and accuracy, we chose to binarize only weights during training while keeping activations in full precision. Binarized weights are well-suited for accumulation-based hardware accelerators such as YodaNN [21] and FINN [22], which replace multiplications with bitwise operations (XNOR + popcount), reducing computation and memory requirements. Previous studies, such as XNOR-Net [23], have shown that binarizing both weights and activations during training severely impacts accuracy, particularly on complex datasets like ImageNet. Keeping activations full-precision during training preserves gradient information and improves accuracy. Activations can still be binarized during inference, ensuring computational efficiency without retraining.

These results demonstrate that AB-Tanh-STE and AB-ArcTan-STE improve the accuracy of BNNs while significantly reducing training time and compu-

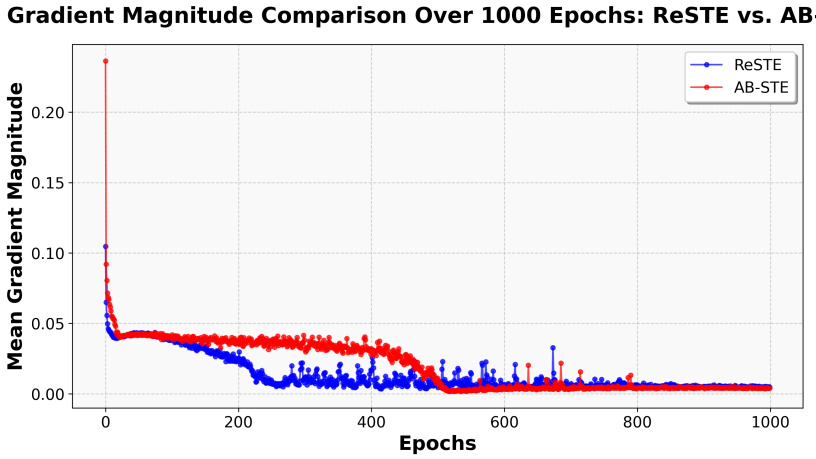


Fig. 4: Training of ResNet-18 with CIFAR-10 dataset

tational overhead, making them well-suited for efficient hardware deployment without sacrificing model performance.

4.5 Gradient Analysis

In Fig. 4, we present a comparison of the mean gradient magnitude across training epochs for ReSTE (blue) and the proposed AB-Tanh-STE (red). The results demonstrate that AB-STE maintains higher gradient magnitudes in the early training phase, ensuring a stronger learning signal and preventing premature convergence. In contrast, ReSTE exhibits a faster decay in gradient magnitudes, potentially limiting the network's ability to explore the optimization landscape effectively during initial training.

As training progresses, AB-STE stabilizes the gradient magnitudes at a consistently higher level compared to ReSTE, facilitating smoother and more structured training dynamics. The gradual decay of gradients in AB-STE ensures that weight updates remain effective, preventing the issue of vanishing gradients commonly observed in deep networks. In contrast, ReSTE experiences notable fluctuations and sharp drops in gradient magnitude, particularly after epoch 200, suggesting less stable weight updates, which could impact model robustness and convergence stability. Beyond epoch 400, AB-STE exhibits lower gradient variance, indicating that it allows for more controlled and adaptive optimization steps. The consistent gradient flow observed in AB-STE contributes to improved training efficiency, ensuring that the network retains sufficient gradient magnitudes for meaningful updates while avoiding instability. On the other hand, ReSTE continues to show irregular oscillations throughout training, making the optimization process less predictable.

The observed improvements in gradient behavior highlight the effectiveness of the proposed AB-STE method in maintaining gradient stability while preserving

representational capacity. By blending linear and non-linear components, AB-STE ensures better gradient flow, reduced gradient saturation, and improved robustness, making it a more effective approach for training binarized deep neural networks in adversarial settings.

5 Conclusion and Discussion

In this work, we introduced two novel Adaptive Blended Straight-Through Estimators (AB-STE): **AB-Tanh-STE** and **AB-ArcTan-STE**, aimed at improving the training of Binary Neural Networks (BNNs). Our approach addresses the critical challenges of gradient instability and inaccurate gradient flow in traditional STE-based methods. By blending linearity and non-linearity, AB-STE maintains both gradient stability and expressiveness, significantly enhancing the overall convergence and training efficiency of BNNs. The extensive experimental evaluation on CIFAR-10 and ImageNet demonstrates that our proposed estimators outperform existing state-of-the-art methods, achieving superior accuracy while preserving training stability. The adaptive nature of our blended estimator provides a flexible mechanism to navigate the challenges of gradient estimation, balancing simplicity and complexity in a manner that improves training outcomes for BNNs. By progressively increasing the non-linearity throughout training, AB-STE effectively mitigates the risks associated with gradient vanishing and exploding, leading to smoother and more stable training dynamics.

Our work also highlights potential directions for future research in **multi-bit quantization-aware training**. The adaptive blending strategy introduced in AB-STE could be extended beyond binary networks to more general quantization schemes, offering a promising pathway to address gradient estimation challenges in multi-bit quantization settings. Moreover, the success of AB-STE in stabilizing gradient flow may provide insights into mitigating the effects of activation quantization during network training, further enhancing the applicability of quantized neural networks to resource-constrained environments. In summary, the proposed AB-STE offers an effective solution to the gradient-related challenges faced by BNNs, and its adaptive blended approach lays a foundation for future advances in quantization-aware training for both binary and multi-bit networks. We believe that this work paves the way for more robust and efficient training methods for neural networks deployed on edge devices, potentially broadening the scope of practical deep learning applications.

Acknowledgment

This research is supported by the German Research Foundation (DFG) under project X-ReAp (380524764) 'X-ReAp: Cross(X)-Layer Runtime Reconfigurable Approximate Architecture.'

References

1. Wu, X.M., Zheng, D., Liu, Z. and Zheng, W.S., 2023. Estimator meets equilibrium perspective: A rectified straight through estimator for binary neural networks training. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 17055-17064).
2. Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2017. ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60(6), pp.84-90.
3. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. and Rabinovich, A., 2015. Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9).
4. Girshick, R., Donahue, J., Darrell, T. and Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 580-587).
5. Li, R., Wang, Y., Liang, F., Qin, H., Yan, J. and Fan, R., 2019. Fully quantized network for object detection. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 2810-2819).
6. Everingham, M., Eslami, S.A., Van Gool, L., Williams, C.K., Winn, J. and Zisserman, A., 2015. The pascal visual object classes challenge: A retrospective. International journal of computer vision, 111, pp.98-136.
7. Courbariaux, M., Bengio, Y. and David, J.P., 2015. Binaryconnect: Training deep neural networks with binary weights during propagations. Advances in neural information processing systems, 28.
8. Courbariaux, M. and Bengio, Y., Binarynet: Training deep neural networks with weights and activations constrained to +1 or -1. arXiv 2016. arXiv preprint arXiv:1602.02830.
9. Wang, Z., Lu, J., Tao, C., Zhou, J. and Tian, Q., 2019. Learning channel-wise interactions for binary convolutional neural networks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 568-577).
10. Jung, S., Son, C., Lee, S., Son, J., Han, J.J., Kwak, Y., Hwang, S.J. and Choi, C., 2019. Learning to quantize deep networks by optimizing quantization intervals with task loss. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 4350-4359).
11. Qin, H., Gong, R., Liu, X., Bai, X., Song, J. and Sebe, N., 2020. Binary neural networks: A survey. Pattern Recognition, 105, p.107281.
12. 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) (pp. 722-737).
13. Tian, Y., Krishnan, D. and Isola, P., 2019. Contrastive representation distillation. arXiv preprint arXiv:1910.10699.
14. Bai, Y., Wang, Y.X. and Liberty, E., 2018. Proxquant: Quantized neural networks via proximal operators. arXiv preprint arXiv:1810.00861.
15. Gong, R., Liu, X., Jiang, S., Li, T., Hu, P., Lin, J., Yu, F. and Yan, J., 2019. Differentiable soft quantization: Bridging full-precision and low-bit neural networks. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 4852-4861).
16. Xu, Y., Han, K., Xu, C., Tang, Y., Xu, C. and Wang, Y., 2021. Learning frequency domain approximation for binary neural networks. Advances in Neural Information Processing Systems, 34, pp.25553-25565.

17. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K. and Fei-Fei, L., 2009, June. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255). Ieee.
18. Krizhevsky, A. and Hinton, G., 2009. Learning multiple layers of features from tiny images.(2009) [online]
19. Qin, H., Gong, R., Liu, X., Shen, M., Wei, Z., Yu, F. and Song, J., 2020. Forward and backward information retention for accurate binary neural networks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 2250-2259).
20. 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.
21. Andri, R., Cavigelli, L., Rossi, D. and Benini, L., 2017. YodaNN: An architecture for ultralow power binary-weight CNN acceleration. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 37(1), pp.48-60.
22. Umuroglu, Y., Fraser, N.J., Gambardella, G., Blott, M., Leong, P., Jahre, M. and Vissers, K., 2017, February. Finn: A framework for fast, scalable binarized neural network inference. In Proceedings of the 2017 ACM/SIGDA international symposium on field-programmable gate arrays (pp. 65-74).
23. Rastegari, M., Ordonez, V., Redmon, J. and Farhadi, A., 2016, September. Xnor-net: Imagenet classification using binary convolutional neural networks. In European conference on computer vision (pp. 525-542). Cham: Springer International Publishing.