BiViT: Extremely Compressed Binary Vision Transformers

Yefei He[1] Zhenyu Lou[1] Luoming Zhang[1] Jing Liu[2] Weijia Wu[1]
Hong Zhou[1†] Bohan Zhuang[2†]

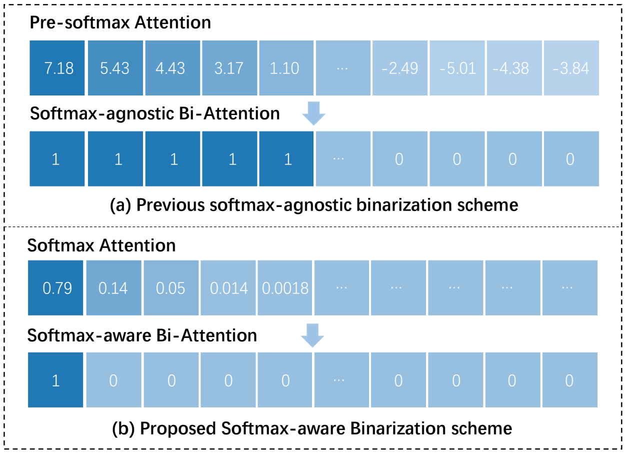
[1]Zhejiang University, China
[2]ZIP Lab, Monash University, Australia

Abstract

*Model binarization can significantly compress model size, reduce energy consumption, and accelerate inference through efficient bit-wise operations. Although binarizing convolutional neural networks have been extensively studied, there is little work on exploring binarization of vision Transformers which underpin most recent breakthroughs in visual recognition. To this end, we propose to solve two fundamental challenges to push the horizon of **Bi**nary **Vi**sion **T**ransformers (BiViT). First, the traditional binary method does not take the long-tailed distribution of softmax attention into consideration, bringing large binarization errors in the attention module. To solve this, we propose Softmax-aware Binarization, which dynamically adapts to the data distribution and reduces the error caused by binarization. Second, to better preserve the information of the pretrained model and restore accuracy, we propose a Cross-layer Binarization scheme that decouples the binarization of self-attention and multi-layer perceptrons (MLPs), and Parameterized Weight Scales which introduce learnable scaling factors for weight binarization. Overall, our method performs favorably against state-of-the-arts by **19.8%** on the TinyImageNet dataset. On ImageNet, our BiViT achieves a competitive **75.6%** Top-1 accuracy over Swin-S model. Additionally, on COCO object detection, our method achieves an mAP of 40.8 with a Swin-T backbone over Cascade Mask R-CNN framework.*

1. Introduction

Vision Transformer (ViT) [21] and its variants have achieved great success in a variety of computer vision tasks, such as image classification [21, 49, 24], object detection [38, 22, 11], semantic segmentation [82, 64, 12], etc. However, massive parameters and calculations of the Transformer models hinder their applications on portable devices such as mobile phones. To tackle the efficiency bottlenecks, various model compression algorithms have been widely studied, such as distillation [65, 67, 33], pruning [58, 84, 75] and quantization [43, 45, 42]. Among them, binary neural networks (BNN) [15, 62, 8] aggressively compress weights and activations to a single bit, which delivers 32× savings on memory consumption, and enables efficient XNOR-popcount bit-wise operations to greatly accelerate model inference and reduce energy consumption.



(a) Previous softmax-agnostic binarization scheme

(b) Proposed Softmax-aware Binarization scheme

Figure 1. **An illustration of attention binarization.** Data is collected from pretrained Nest-T model and "Bi-Attention" denotes for binarized attentions. (a) Pre-softmax attention binarized by BiBERT [60]. (b) Softmax attention binarized by our method.

However, the performance degradation restricts the wide application of BNNs, which is mainly caused by the limited representational ability and difficulty in optimization. To improve the performance of BNNs, binarized convolutional neural networks (CNNs) literature has been extensively studied to design accurate binarization functions [62, 83, 8], enhance the representation ability [54, 52, 86] and relieve the gradient approximation error in optimization [61, 30, 4]. Also, many attempts have been made in previous studies to binarize BERT [19] for natural language processing (NLP) tasks, such as calibrating the attention value range mismatch [60], customizing knowledge distillation and techniques in binary CNNs to Transformers [50].

[†] H. Zhou and B. Zhuang are corresponding authors.

However, there are few studies on the binarization of ViTs so far. Therefore, it is highly critical and imperative to explore BiViT for diverse edge devices to infer ViT at low-latency and low-power for real-world flexibility.

In addition to the common challenges mentioned above, binarizing Transformers presents two new exclusive challenges. *Firstly, it lacks effective methods for accurately binarizing softmax attention.* Self-attention module aims to encode pairwise similarity between all the tokens [69], which is very different from convolutional or fully-connected layers. Specifically, the values of attention scores are all positive values between (0, 1) and exhibit long-tailed distributions after $\mathrm{Softmax}$ operation (See Figures 1 (b) and 3). In contrast, the ordinary weights have both positive and negative values and follow a bell-shaped distribution. Moreover, attention scores are generated during inference while the ordinary weights are fixed after finishing training. Consequently, the functionality and data distribution of softmax attention differ significantly from ordinary weights. As to this problem, the recent study BiBERT [60] proposes to maximize the information entropy of binary attention scores by applying Bool function on pre-softmax attentions, resulting in the balanced number of zeros and ones, as shown in Figure 1 (a). However, the softmax attention scores are actually dominated by few elements, thus the number of ones in binary attentions should be much less than the number of zeros to ensure a low quantization error (see Figure 1 (b)). In other words, BiBERT follows a softmax-agnostic approach and overlooks the effect of $\mathrm{Softmax}$ on the distribution, resulting in mismatched attention score distributions before and after binarization and leading to significant quantization errors (See Table 1). Another study BiT [50] proposes to learn both scales and thresholds for weight and attention binarization, making them fixed during inference. While this method works well for weight binarization, it neglects the dynamics of attention scores and cannot adapt well to the changing distribution of attentions, as this approach remains softmax-agnostic at inference time as well.

Secondly, how to preserve the information in the pretrained ViTs during binarization is under explored. Unlike binary CNNs that perform well when training from scratch [61, 68, 52], we observe that BiViTs heavily rely on pretrained models and are sensitive to quantization, as shown in Figure 2. Even if the initial weights are derived from the pretrained model, directly binarizing all parameters still causes a huge loss of pretrained information, which then leads to a severe performance drop. Also, the loss of pretrained information is difficult for Transformers to recover through quantization-aware training (QAT). In particular, MLP modules account for nearly half of the computations and parameters within a Transformer [47]. They are mainly composed of 1×1 convolutions, which are widely recognized to be difficult to binarize due to the limited representational capability [86, 23, 7]. Therefore, the effective binarization of softmax attention and the retention of information from pretrained models remain open questions.

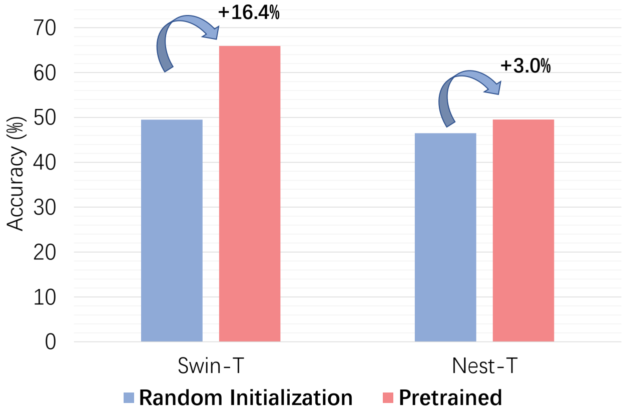


Figure 2. **Impact of pretrained model when binarizing Transformers.** The experiment is conducted on TinyImageNet dataset. Initiating Transformers from the pretrained models greatly boosts the accuracy.

To reduce the quantization error in binarizing attentions, we first analyze the long-tailed distribution of softmax attention scores and discover their differing patterns across different attention vectors. To adaptively search the optimal thresholds for binarization, we propose an optimization algorithm based on sparse coding and coordinate descent, and further propose an efficient approximation called Softmax-aware Binarization (SAB) to avoid conducting the optimization on each forward pass. Moreover, to retain pretrained information and further enhance the model representational capability, we then propose Cross-layer Binarization (CLB) to decouple the quantization of self-attention and MLPs to avoid mutual interference and introduce Parameterized Weight Scales (PWS) for weights binarization. To our best knowledge, we are the pioneering work to probe binarizing Transformers for vision tasks.

In summary, our contributions are as follows:

- We are the pioneering work that explores binary vision Transformers, a demanding recipe for efficient ViT inference.
- We design a Softmax-aware Binarization scheme for the self-attention module, which adapts to the long-tailed attention scores distribution and greatly reduces the quantization error.
- We propose Cross-layer Binarization and Parameterized Weight Scales to retain pretrained information and further enhance the representational ability of BiViTs, improving convergence and accuracy.
- Experiments on TinyImageNet and ImageNet for image classification, and COCO for object detection, demonstrate that it consistently outperforms current

state-of-the-arts by large margins, serving as a strong baseline for future research.

2. Related Work

2.1. Vision Transformers

Transformer [69] is initially proposed to process long sequences in NLP tasks. ViT [21] first adapts Transformers to vision tasks by splitting images into patches and processing them as token sequences. DeiT [66] further improves the data efficiency of vision Transformers. Benefiting from the global receptive field and the powerful long-range modeling capabilities of self-attention, ViT demonstrates promising performance against CNN counterparts. Many follow-up works are proposed to explore hierarchical structures [49, 70, 81], insert the convolutional inductive bias [41, 25, 17] and apply ViTs to various vision tasks [80, 77, 22]. However, the inference speed of ViTs is generally slower than that of CNNs in practical applications [40]. The reasons mainly include the lack of special optimization (such as Winograd [48] for convolutional layers) and the quadratic computational complexity of the self-attention module. To reduce the computational complexity of ViTs, many methods have been proposed, including linear complexity attention [63, 9, 71], network pruning [31, 74, 13] and quantization [43, 45, 53]. However, current Transformer quantization literature mainly focuses on fixed-point quantization, either through Quantization-Aware Training (QAT) [39, 43] or Post-Training Quantization (PTQ) [53, 45, 76]. Research on ternary or binary quantization remains to be studied.

2.2. Binary Neural Networks

BNNs seek to quantize both weights and activations to 1-bit, which greatly reduces the complexity of the model. The binarization of models usually requires QAT to restore accuracy. To overcome the non-differentiability of quantizer during training and the limited representational capacity, many methods have been proposed to help binarize CNNs, such as binary-friendly model structures [54, 46, 85, 57, 7, 6], knowledge distillation [52, 56, 55], gradient approximation [28, 61, 79, 20], optimizer selection [51, 1, 16], *etc*. Although some of them are also effective for Transformer models, as analyzed in BiT [50], methods targeting on binary Transformers still need to be developed to relieve accuracy degradation.

The literature closely related to our work includes BinaryBERT [3], BiBERT [60] and BiT [50]. They propose techniques to improve the binarization of BERT [19] variants, such as customized binarization functions and model distillation, and evaluate them on NLP tasks. However, none of these methods are evaluated on vision tasks. In the following sections, we will migrate these methods to DeiT [66], Swin [49] and NesT [81] as our baselines to test their performance and analyze the key challenges. Then we propose to improve BiViT's performance with our SAB, CLB and PWS. To the best of our knowledge, we are the pioneering work studying BiViTs.

3. Method

3.1. Preliminaries

Generally, standard BNNs follow [62] to use Sign function to binarize weights and activations to {-1, +1}, and exploit Straight-Through Estimator (STE) [5] to overcome the non-differentiability of the Sign function, as follows:

$$\hat{x} = \mathrm{Sign}(x) = \begin{cases} +1, \text{if } x \geq 0 \\ -1, \text{otherwise}, \end{cases} \tag{1}$$

$$\frac{\partial \mathcal{L}}{\partial x} \approx \begin{cases} \frac{\partial \mathcal{L}}{\partial \hat{x}}, \text{if } |x| \leq 1 \\ \quad 0, \text{otherwise}. \end{cases} \tag{2}$$

To approximate the full-precision $\mathbf{x} \in \mathbb{R}^n$, BNNs further introduce a scaling factor $\alpha \in \mathbb{R}^+$ to reduce the quantization error:

$$\alpha = \frac{\|\mathbf{x}\|_{\ell_1}}{n}, \quad \mathbf{x} \approx \alpha\hat{\mathbf{x}}. \tag{3}$$

However, binarization using Sign can be problematic in Transformers and be very different from CNNs. Specifically, in the self-attention mechanism [69], attention is calculated as

$$\mathrm{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathrm{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}, \tag{4}$$

where $\mathbf{Q}$, $\mathbf{K}$, $\mathbf{V}$ are query, key and value matrics respectively, and d_k is the dimension of the key. In the following, we will denote softmax attention vector (*i.e.*, one row of the softmax attention matrix) as $\mathbf{a}_s$ and pre-softmax attention vector as $\mathbf{a}_p$.

According to the definition of Softmax, the results are non-negative and they will all be $+1$ after Sign function. In order to solve the problem of value range mismatch, BiBERT [60] proposes to use the Bool function to binarize pre-softmax attention scores to $\{0, 1\}$, which is defined as:

$$\mathrm{Bool}(a_p) = \begin{cases} 1, \text{if } a_p \geq 0 \\ 0, \text{otherwise}. \end{cases} \tag{5}$$

However, this will lead to huge quantization errors since Softmax is totally discarded, as we will analyze in Section 3.2.

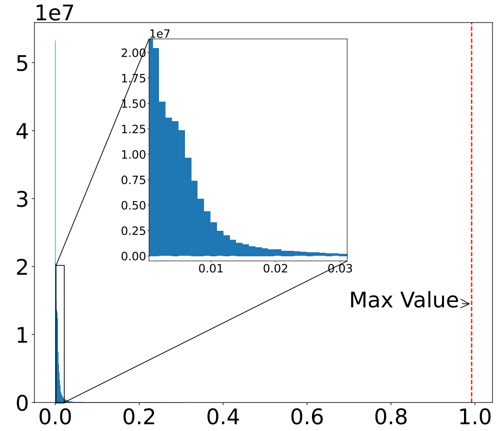


Figure 3. **The long-tailed distribution of attention scores.** Most attention scores are around zero but the maximum values can reach 0.99.

3.2. Softmax-aware Binarization

Self-attention is designed to model global relationships among different patches (tokens) and focuses on important token pairs. Figure 3 presents the distribution of attention scores in the pretrained NesT-T [81] model. We observe that after $\mathrm{Softmax}$ operation, attention scores follow a long-tailed distribution and more than 99.5% of them are less than 0.05, which is highly sparse. To further investigate this distribution, we take a deep look at an actual attention vector from the NesT-T pretrained model. As shown in Figure 1 (a), if we follow BiBERT [60] and use Bool function to binarize pre-softmax attentions, nearly half of the attention scores are set to 1, indicating they have the same contribution to binarization, which is inconsistent with the actual distribution of softmax attention scores where only few values dominate (see Figure 1 (b)). BiT [50] proposes learnable thresholds for binarization but it is fixed after training, while distributions of attention scores can vary with different input images and may differ between each attention vector, leading to a suboptimal solution.

In order to reduce the quantization error while binarizing softmax attention scores, we argue that the ideal binarization method should satisfy the following two properties: 1) The proportion of activated scores (set to 1) in binary attentions should be smaller compared to directly using the Bool function. As shown in Figure 3, most values are around 0, which barely contributes to the result during calculation, and only a few significant values are considered. 2) The activation thresholds should not be a fixed value. $\mathrm{Softmax}$ is operated on every row-wise attention vector and different softmax attention vectors follow distinct distributions. For example, the maximum value of some of them can reach 0.99, while the others are only about 0.05. Empirically, even though most softmax attentions are dominated by only a few elements, the thresholds to activate should be different across all attention vectors.

To achieve this, the key is to find the optimal threshold T^* for binarizing each softmax attention vector (*i.e.*, T^* is different for each row). Inspired by sparse coding [36] and LQ-Nets [78], we formulate the quantized attention vector $\mathbf{a}_q \in \mathbb{R}^n$ by the inner product between a basis vector $\mathbf{v} \in \mathbb{R}^k$ and the binary encoding vector $\mathbf{b} \in \{0,1\}^{k\times n}$:

$$\mathbf{a}_q = \mathbf{v}^T\mathbf{b}, \tag{6}$$

where k is the target bitwidth. Then the optimization problem can be formulated as:

$$\mathbf{v}^*, \mathbf{b}^* = \underset{\mathbf{v},\mathbf{b}}{\operatorname{argmin}} \|\mathbf{v}^T\mathbf{b} - \mathbf{a}_s\|_2^2, \quad s.t.\ \mathbf{b} \in \{0,1\}^{k\times n}. \tag{7}$$

In this paper, the bitwidth k is set to 1, thus the basis vector $\mathbf{v}$ becomes a scalar v. However, with both v and $\mathbf{b}$ to be solved, brute-force search can be computationally expensive. Instead, the optimization problem can be efficiently solved in a coordinate descent approach. Specifically, we alternatively optimize the basis v and binary encoding vector $\mathbf{b}$ while keeping another fixed:

Update v: With the fixed binary encoding vector $\mathbf{b}$, the optimization problem will degenerate to a special case of linear regression. Therefore, the optimal v can be derived by:

$$v^* = \frac{\mathbf{a}_s \cdot \mathbf{b}}{\|\mathbf{b}\|_2^2}, \tag{8}$$

where $\cdot$ represents the dot product between two vectors. **Update $\mathbf{b}$:** Since we get the optimal v with Eq. (8), the two values for binarization becomes $\{0, v^*\}$. Then the optimal threshold can be simply calculated as:

$$T^* = \frac{0 + v^*}{2}. \tag{9}$$

Then we binarize the full-precision softmax attention vector with the threshold T^* to update the binary encoding $\mathbf{b}$:

$$\mathbf{b}^* = \mathrm{Bool}(\mathbf{a}_s - T^*). \tag{10}$$

The coordinate descent optimization process is summarized in Algorithm 1. After N iterations, the quantization error between binary attentions and full-precision softmax attentions decreases significantly, as shown in the second row of Table 1.

Table 1. Quantization error under different methods. We set $N = 5$ in practice.

Method	Quantization Error
BiBERT	0.683
Optimal T^*	2.58e-05
Approximate T	2.72e-05
Approximate T w/o scales	0.141

Algorithm 1 The coordinate descent optimization.

```
Input: softmax attention vector $\mathbf{a}_s$
Output: the basis scalar $v$, the binary encoding $\mathbf{b}$
Procedure:
    Initialize $\mathbf{b}^{(0)} = \mathrm{Sign}(\mathbf{a}_s)$
    for $t = 1 \rightarrow N$ do
        Update $v^{(t)}$ with $\mathbf{a}_s$ and $\mathbf{b}^{(t-1)}$ per Eq. (8)
        Update $\mathbf{b}^{(t)}$ with $\mathbf{a}_s$ and $v^{(t)}$ per Eqs. (9) and (10)
    end for
```

Although this optimization strategy minimizes the quantization error, it is not practical to optimize each generated attention vector during inference. Besides, we calculate an optimal v^* for each attention vector, which introduces an extra computational burden.

To simplify the optimization process and reduce computational complexity, we seek to establish a relationship between the optimal thresholds T^* (calculated by Eq. (9)) and the distribution of the softmax attention scores. In practice, we sample 128 images from ImageNet [18] dataset, obtain 10^4 attention vectors and calculate the corresponding optimal thresholds T^* with the proposed method. As illustrated in Figure 4, we observe that the optimal thresholds T^* show a strong correlation with the maximum values of the softmax attention vectors. Therefore, we estimate a coefficient β via linear regression with these sampled data to approximate thresholds T for each attention vector and thus accelerate the optimization:

$$T = \beta \mathrm{Max}(\mathbf{a}_s). \tag{11}$$

Experimental result demonstrates that this approximation barely increases the quantization error, as shown in the third row of Table 1.

However, compared with the previous methods [60, 3, 50], multiplying the basis scalar v by the binary encoding vector $\mathbf{b}$ to get $\mathbf{a}_q$ (as defined in Eq. (6)) for each attention vector still introduces extra computation. To keep the same computational complexity as previous methods, we make a second approximation and further discard the basis scalar v. In this case, the quantization error is shown in the fourth row of Table 1. It should be noted that this step is simply a trade-off between accuracy and complexity. By default, we use the algorithm with two approximations for experiments.

Note that there are also many intuitive schemes for approximating the $\mathrm{Softmax}$ operation, such as Top-N algorithm with a learnable parameter N and learnable thresholds T proposed in BiT [50]. For Top-N algorithm, it can take heavy computations to obtain N and needs to find N maximum values during inference. Instead, we only need to find the maximum value through the proposed twice approximation algorithm, which is faster and barely increases error. Compared with BiT [50], our method calculates the thresholds dynamically and can adapt well to the attention distributions during inference.

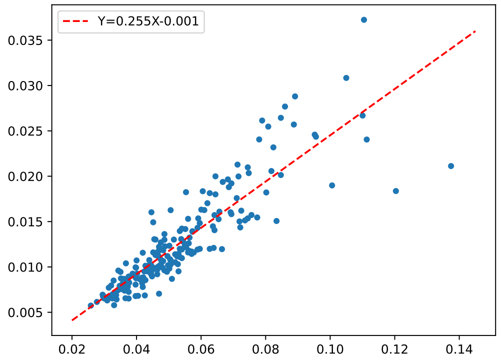


Figure 4. **Relation between the maximum values of softmax attention vectors (X-axis) and optimal thresholds T^* (Y-axis).** Blue dots are the maximum values of softmax attention vectors sampled from the pretrained NesT-T model and red dashed line represents the result of linear regression on these attention scores. For simplicity, we remove the bias term in the following analysis.

In the backward pass, BiBERT employs the STE to propagate gradients from binary attentions $\mathbf{a}_q$ to pre-softmax attentions $\mathbf{a}_p$ *without* considering the $\mathrm{Softmax}$ operation:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{a}_p} \approx \frac{\partial \mathcal{L}}{\partial \mathbf{a}_q}. \tag{12}$$

In contrast, our proposed method employs the STE to propagate gradients from binary attentions $\mathbf{a}_q$ to softmax attentions $\mathbf{a}_s$, making it *Softmax-aware* during backpropagation:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{a}_p} = \frac{\partial \mathcal{L}}{\partial \mathbf{a}_s}\frac{\partial \mathbf{a}_s}{\partial \mathbf{a}_p} \approx \frac{\partial \mathcal{L}}{\partial \mathbf{a}_q}\frac{\partial \mathbf{a}_s}{\partial \mathbf{a}_p}. \tag{13}$$

We will further show in Section 4.4.1 that this greatly helps the training of BiViTs.

Overall, the training process of the proposed SAB is summarized in Algorithm 2.

Algorithm 2 SAB for self-attention modules

```
Input: softmax attention scores $\mathbf{a}_s \in \mathbb{R}^n$
Output: binary attention scores $\mathbf{a}_q \in \mathbb{R}^n$
Forward propagation:
    Approximate the thresholds $T$ by the maximum value of attentions with Eq. (11).
    Binarize attentions with thresholds $T$ by Eq. (10).
Backward propagation:
    Calculate the gradients w.r.t. $\mathbf{a}_s$ with Eq. (??).
```

3.3. Information Preservation

3.3.1 Cross-layer Binarization

Compared with binary BERT [60, 50, 3], we find that BiViTs are more difficult to optimize. To justify this, we directly migrate BiBERT [60] to Swin-T [49] and NesT-T [81] to evaluate its performance on image classification tasks. The results are shown in Table 2. We observe that its accuracy degradation in image classification tasks can reach

40% on TinyImageNet dataset, which indicates that vanilla BiViTs cannot make good use of the pretrained information and is difficult to optimize on vision tasks.

To explore the reasons, we present the architecture and parameters of Swin-T as an example in Figure 5. As analyzed in Section 1, the MLP modules are hard to quantize due to the limited representational capability of 1×1 convolutions and have more parameters than the self-attention module. To tackle this problem, we propose Cross-layer Binarization (CLB), which is analogous to the previous two-step training scheme [87, 55], to decouple the quantization of self-attention module and MLP module to reduce mutual interference. Specifically, in the first stage, we keep MLPs to full precision and binarize all the self-attention modules with our SAB scheme. Then in the second stage, we binarize MLPs to get the final model.

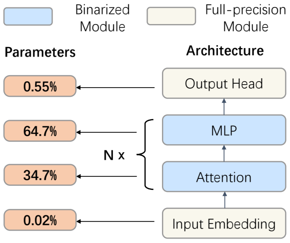


Figure 5. **The architecture and parameters of Swin-T model.** MLP is difficult to binarize due to limited representational capacity and has far more parameters than other modules.

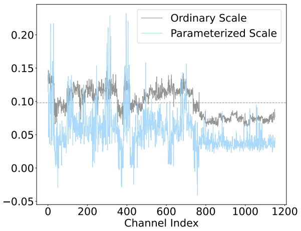


Figure 6. **Ordinary and parameterized scaling factors in NesT-T model.** Data are collected from binarized Nest-T model. The dashed line represents the mean value of ordinary scaling factors.

Compared with previous two-step training schemes that first binarize activations and then weights [55, 52, 2], CLB is designed for Transformers to relieve the mutual interference and mitigate information loss mainly caused by binarizing MLPs.

The experimental results demonstrate that using CLB brings more accuracy improvement than the traditional two-step training scheme, as we will show in Section 4.4.2.

3.3.2 Parameterized Weight Scales

To further narrow the performance gap between the binarized model and the full-precision counterpart, an intuitive idea is to increase the representation ability of the binarized model. However, this usually results in increased computational complexity.

Motivated by SE-Net [32], channel-wise scaling factors can be regarded as the importance of each channel, rather than an approximation of its distribution. In order to preserve the model structure and complexity while enhancing its representation ability, we directly replace ordinary scaling factors (as defined in Eq. (3)) with learnable parameters. The parameterized scaling factors could be optimized in conjunction with other network parameters via backward propagation during training. As shown in Figure 6, the deviation of ordinary scaling factors across channels is small, indicating that the weight distribution of each channel is similar. In contrast, the parameterized scaling factors vary greatly from channel to channel, showing that it learns to pay more attention to specific channels and thus enhancing the representational capacity of the model.

4. Experiments on Image Classification

4.1. Implementation Details

Datasets and architectures. We conduct image classification experiments on two standard benchmarks: TinyImageNet [73] and ImageNet (ILSVRC12) [18]. The input resolution is 224×224. For data augmentation, we follow the settings in DeiT [66], which are common practices in ViTs. To demonstrate the versatility of our method, we adopt three widely-used efficient architectures: DeiT [66], Swin [49] and NesT [81]. All the blocks in Transformer models are binarized without exception. For binary attention modules, all weights and intermediate results including $\mathbf{Q}$, $\mathbf{K}$, $\mathbf{V}$ and projection layers, are binarized. For binary MLP modules, weights are binarized in all experiments. We leave the input embedding layer and output layer unbinarized as it is the common practice of BNNs [62]. The binary operations (BOPs) and floating-point operations (FLOPs) are counted separately and the reported total operations (OPs) are calculated by $\mathrm{OPs} = \mathrm{BOPs}/64 + \mathrm{FLOPs}$, following [62, 52].

Training setup. All experiments are implemented with PyTorch [59] and Timm [72] library. The iteration number for coordinate descent is $N = 5$. For both datasets, we employ Adam [34] optimizers without weight decay and train models for 300 epochs using a cosine annealing schedule with 5 epochs of warm-up. The initial learning rate is set to 5e-4. When training is split into two stages, we train 150 epochs at each stage to keep the number of total iterations the same. Knowledge distillation (KD) [29] is used in all experiments. Specifically, we use the distribution loss proposed in [52] for optimization. Before training, all parameters are initialized with the pretrained model.

4.2. Comparison with SOTA methods

4.2.1 Evaluation on TinyImgnet

We begin our evaluation on the TinyImageNet, and the results are presented in Table 2. Previous Transformer binarization methods [50, 60] exhibit significant accuracy degradation, which limits their practicality. Remarkably, in some cases, the performance of BiT [50] can be worse than BiBERT [60], indicating that the learned thresholds T may not be suitable for softmax attentions during inference or can even be detrimental. By contrast, our proposed SAB

Table 2. Comparisons of different network architectures on Tiny-ImageNet. Here, "FP" means full-precision pretrained model, "ATTN" denotes attention module and "W/A" represents the number of bits used in weights or activations.

Model	Method	ATTN Bitwidth (W/A)	MLP Bitwidth (W/A)	Top-1 Acc. (%)
Swin-T	FP	32/32	32/32	80.57
	BiBERT	1/1	1/1	41.89
	BiT	1/1	1/1	40.52
	Ours	1/1	1/1	**58.66**
	FP	32/32	32/32	80.57
	BiBERT	1/1	1/32	65.93
	BiT	1/1	1/32	61.82
	Ours	1/1	1/32	**71.20**
NesT-T	FP	32/32	32/32	80.31
	BiBERT	1/1	1/1	32.39
	BiT	1/1	1/1	34.72
	Ours	1/1	1/1	**52.21**
	FP	32/32	32/32	80.31
	BiBERT	1/1	1/32	49.53
	BiT	1/1	1/32	46.43
	Ours	1/1	1/32	**69.83**

method dynamically determines the thresholds according to the distribution of softmax attention. For models with all weights and activations binarized, our approach can boost the Top-1 accuracy by almost 20% (52.21% vs. 32.39% for NesT-T), greatly narrowing the performance gap between binary and full-precision models. However, we still observed a large accuracy drop, which can be attributed to the significant contribution of MLP modules to the parameters and their limited representational capability. To enhance practicality, we keep the activations in MLP modules as full-precision.

The experimental results demonstrate that preserving activations as full-precision can significantly mitigate the performance degradation caused by fully-binarized MLPs (69.83% vs. 52.21% for NesT-T), achieving a better trade-off between accuracy and complexity. In this case, the model size can still be compressed by 32×, and the costly multiply-accumulate operations can be replaced by cheap accumulations. With this configuration, our method also outperforms previous methods BiBERT and BiT by large margins, with improvements of up to 20.3% for Nest-T.

We also found that binary Swin-T outperforms binary NesT-T, while their full-precision models perform similarly. This difference may be attributed to the mask mechanism in the local window attention of Swin, which limits the number of elements greater than zero in the pre-softmax attention, thereby aiding in model binarization. Nevertheless, our method provides substantial accuracy gains for most Transformers with standard self-attention (like NesT).

4.2.2 Evaluation on ImageNet

We further evaluate the effectiveness of our method on the ImageNet, and the results are shown in Table 3. To preserve the accuracy, we binarize all weights while leaving the activations in MLP modules full-precision. In all cases, our method obtains the best performance. For full-attention models (like DeiT), the accuracy is significantly lower than local-attention models, especially when the model capacity is insufficient (such as DeiT-T). We speculate that the inductive bias of local information is very important for highly compressed BiViTs. For Swin-T, our method outperforms previous SOTA by a margin of 2.5%. For NesT-T, where the previous method even fails to converge, our method obtains a 68.7% Top-1 accuracy. We also conduct experiments on larger ViTs like DeiT-B and Swin-S, achieving a highly competitive accuracy of up to 75.6% over Swin-S. These results demonstrate the feasibility of BiViTs in visual tasks for the first time.

Table 3. Performance comparisons of different architectures on ImageNet. "*" denotes the model fails to converge.

Base Arch	Model	Method	Size (MB)	OPs (10ˆ9)	Top-1 Acc.(%)
CNN	ResNet-18	FP	46.8	1.83	69.6
		AdaBin[1]	5.56	0.18	63.1
		IR-Net[2]	5.56	0.99	66.5
	ResNet-34	FP	87.2	3.68	73.3
		AdaBin[1]	8.12	0.21	66.4
		IR-Net[2]	8.12	2.11	70.4
Transformer	PVTv2-B1	FP	53.5	2.12	78.8
		BiBERT	11.3	0.95	64.7
		Ours	11.3	0.95	**67.3**
	CvT-13	FP	76.6	4.59	81.4
		BiBERT	7.11	1.73	55.4
		Ours	7.11	1.73	**64.9**
	DeiT-T	FP	22.8	1.26	71.7
		BiBERT	2.22	0.39	25.4
		Ours	2.22	0.39	**37.9**
	DeiT-B	FP	346.3	17.6	81.8
		BiBERT	16.8	5.81	67.5
		Ours	16.8	5.81	**69.6**
	Swin-T	FP	113.2	4.51	81.2
		BiBERT	12.6	1.62	68.3
		Ours	12.6	1.62	**70.8**
	Swin-S	FP	198.4	8.77	83.2
		BiBERT	15.4	3.04	74.0
		Ours	15.4	3.04	**75.6**
	Nest-T	FP	68.4	5.83	81.1
		BiBERT	8.96	2.49	0.27*
		Ours	8.96	2.49	**68.7**
	Nest-S	FP	153.4	10.4	83.3
		BiBERT	11.7	3.92	73.5
		Ours	11.7	3.92	**74.9**

4.3. Comparison with Binary CNNs

We present a comparison of the model size, total operations, and accuracy between binary ResNet [27] and BiViTs, as shown in Table 3. CNNs with their convolutional inductive bias and fewer 1×1 convolution layers can achieve good performance with minimal parameters and operations, surpassing similarly sized BiViTs such as DeiT-T. However, as model capacity increases, the advantages of Transformers become more apparent. The global receptive field and

[1] CNNs with binary weights and activations.

[2] CNNs with binary weights and full-precision activations.

attention mechanism in Transformers provide strong representational capability, resulting in higher accuracy for full-precision ViT models than ResNet models with similar parameters. As the information of the pretrained model is inherited, a stronger teacher model can significantly improve BiViT's training. For instance, binary Swin-S has similar OPs as full-precision ResNet-34, yet it achieves better accuracy (75.6% vs. 73.3%) and reduces model size by 5.66×. Moreover, we measure the latency of matrix multiplication operations in BiViTs and full-precision models using an RTX3090 GPU, as shown in Table 4. For floating-point (FP) operations, we utilize cuDNN [14], while binary operations are implemented using BTC-BNN [37]. In our default setting where attention modules are binarized while the MLP modules retain FP activations, we observe a 1.99× reduction in latency compared to its full-precision counterpart on Swin-T. Moreover, when we fully binarize the MLP modules as well, the speedup further increases to 4.39×. However, due to the long-standing optimization of convolution implementations, such as Winograd [35], the current latency of BiViTs is slightly higher than that of FP CNNs with similar OPs, such as binary Swin-S model and FP ResNet-34.

Table 4. Latency comparisons with different models on ImageNet.

Model	Method	Size (MB)	OPs (10^9)	Latency (ms)	Top-1 Acc. (%)
ResNet-18	FP	46.8	1.83	0.91	69.6
ResNet-34	FP	87.2	3.68	2.46	73.3
Swin-T	FP	113.2	4.51	3.03	81.2
	Ours[3]	12.6	1.62	1.52	70.8
	Ours[4]	12.6	0.27	0.69	-
Swin-S	FP	198.4	8.77	5.42	83.2
	Ours[3]	15.4	3.04	2.91	75.6
	Ours[4]	15.4	0.34	1.16	-
Nest-T	FP	68.4	5.83	3.50	81.1
	Ours[3]	8.96	2.49	2.08	68.7
	Ours[4]	8.96	1.12	1.21	-
Nest-S	FP	153.4	10.4	5.88	83.3
	Ours[3]	11.7	3.92	3.46	74.0
	Ours[4]	11.7	1.19	1.70	-

4.4. Ablation Studies

4.4.1 Effect of Softmax-aware Binarization

First, we conducted ablation experiments over Swin-T and NesT-T models on TinyImageNet to prove the effectiveness of the proposed SAB scheme. To eliminate the impact of MLPs, we keep them full-precision and only binarize attention modules. As shown in Table 5, our SAB consistently narrows the accuracy gap between the full-precision teacher and the binary student network, indicating successful suppression of the quantization error in the self-attention module. Incorporating $\mathrm{Softmax}$ for gradient approximation brings an accuracy improvement of 2.02%, because it mitigates the mismatch issue between backpropagation using STE and the $\mathrm{Softmax}$ operation in the forward pass.

Table 5. Ablation study on Softmax-aware Binarization. † denotes the network not considering $\mathrm{Softmax}$ in the backward pass.

Model	Method	ATTN Bitwidth (W/A)	Top-1 Acc. (%)
Swin-T	FP	32/32	80.57
	BiBERT	1/1	73.39
	+ SAB	1/1	**74.62**
NesT-T	FP	32/32	80.31
	BiBERT	1/1	68.51
	+ SAB†	1/1	68.71
	+ SAB	1/1	**70.73**

Table 6. Comparisons of binary attention's performance under different thresholds. The experiment is conducted over NesT-T model on TinyImageNet.

Method	Top-1 Acc. (%)
FP	80.31
BiBERT	68.51
Ours (β=0.20)	69.18
Ours (β=0.25)	70.73
Ours (β=0.35)	70.81
Ours (β=0.45)	70.68

Also, we conduct experiments to verify the impact of β estimation (see Eq. (11)) on the accuracy of the model. The results in Table 6 show that the model is not sensitive to β within a reasonable interval (about 0.25 to 0.45). By default, β is set to 0.25 to enable efficient bit-shift operation in Eq. (11). However, the accuracy of the model decreases when β is too small (less than 0.2). This indicates that as the thresholds become too small, our SAB is less effective and too many attention scores are activated.

4.4.2 Effect of Information Preservation

To demonstrate the effectiveness of our proposed CLB training scheme, we conducted experiments comparing the accuracy of binary Nest-T trained with CLB to those trained using traditional two-step schemes [55], as well as those trained directly with one-step training scheme on TinyImageNet. The activations in MLP modules remain full-precision. The results, shown in Figure 7 and Table 7, demonstrate that using CLB can accelerate the convergence process of the BiViTs and significantly improve the accuracy by 11.6% compared to one-step training. Furthermore, CLB outperforms the conventional two-step training scheme by 4.1%, highlighting its strong ability to retain pretrained information. We expect that our CLB and the conventional two-step training methods can be combined to further improve the accuracy since they are orthogonal, but will leave it for future work.

For PWS, it can be applied to both self-attention and MLP modules. To show the improvement brought by PWS, we conduct experiments starting with a BiBERT-

[3] Transformers with binary weights, binary activations within attentions and full-precision activations within MLPs.

[4] Transformers with binary weights and activations.

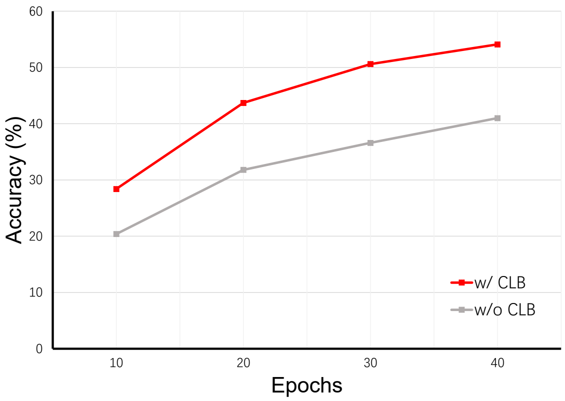


Figure 7. **Training accuracy curves with less training epochs.** The experiment is conducted over NesT-T model on TineImageNet.

Table 7. Ablation study on Cross-layer Binarization. Here, "TS" denotes traditional two-step training scheme.

Method	ATTN Bitwidth (W/A)	MLP Bitwidth (W/A)	Top-1 Acc. (%)
FP	32/32	32/32	80.31
Ours (w/o CLB)	1/1	1/32	58.20
Ours (w/ TS)	1/1	1/32	**65.64**
Ours (w/ CLB)	1/1	1/32	**69.83**

based baseline and then changing scaling factors to learnable parameters in self-attention and MLP modules separately. As shown in Table 8, PWS is effective in both modules, resulting in an accuracy improvement of 4.2% and 1.3%, respectively. Therefore, we use PWS in both modules by default if they are binarized.

Table 8. Ablation study on Parameterized Weight Scales. The experiment is conducted over Nest-T model on TinyImageNet.

Method	ATTN Bitwidth (W/A)	MLP Bitwidth (W/A)	Top-1 (%)
FP	32/32	32/32	80.31
BiBERT	1/1	32/32	68.51
+PWS	1/1	32/32	**72.75**
FP	32/32	32/32	80.31
BiBERT	32/32	1/1	70.02
+PWS	32/32	1/1	**71.35**

5. Experiments on Object Detection

In this section, we present the results of object detection and instance segmentation experiments over Swin-T on COCO 2017 [44] validation set. Currently, we only binarize the self-attention modules in the backbone. The experiments are implemented with classic object detection frameworks Mask R-CNN [26] and Cascade Mask R-CNN [10]. For training strategy and hyper-parameters, we follow the implementation in Swin [49], which takes ImageNet pretrained model as initialization and only trains for 12 epochs.

Table 9 compares the results of different binarization methods on both frameworks. When evaluated with the Mask R-CNN framework, the proposed method improves the performance over BiBERT by 1.4% and 1.2% mAP on the object detection and instance segmentation tasks, respectively. For the Cascade Mask R-CNN framework, the performance improvement brought by our method is even greater, achieving a competitive 40.8% mAP on object detection. More results on COCO are available in the supplementary material.

Table 9. Comparisons of different methods and backbones on COCO 2017 validation set. Here, "FP" represents the full-precision pretrained model.

Framework	Method	Task	AP	AP_{50}	AP_{75}
Mask R-CNN	FP	Object Detection	43.7	66.6	47.7
	BiBERT		32.0	53.9	33.7
	Ours		**33.4**	**55.0**	**35.2**
	FP	Instance Segmentation	39.8	63.3	42.7
	BiBERT		30.4	51.0	31.9
	Ours		**31.6**	**51.7**	**33.4**
Cascade Mask R-CNN	FP	Object Detection	48.1	67.1	52.2
	BiBERT		39.2	57.3	42.5
	Ours		**40.8**	**59.2**	**44.1**
	FP	Instance Segmentation	41.7	64.4	45.0
	BiBERT		34.5	54.5	36.8
	Ours		**35.7**	**56.5**	**38.2**

6. Conclusion

In this paper, we have proposed to tackle two fundamental challenges with customized solutions for BiViTs, and have successfully applied BiViTs to visual tasks for the first time. To deal with the long-tailed distribution of softmax attention, we have proposed the Softmax-aware Binarization for self-attention, the core module of Transformers, which greatly reduces the quantization error. To preserve information from pretrained model, we have proposed the Cross-layer Binarization scheme that decouples the quantization of self-attention and MLPs. Moreover, we have introduced Parameterized Weight Scales to enhance representational ability of BiViTs. Our BiViT has achieved significant accuracy improvement over SOTA on the image classification task, with up to 75.6% Top-1 accuracy on ImageNet over Swin-S model. We have also conducted extensive experiments on COCO object detection and instance segmentation, demonstrating that BiViTs can be extended to downstream tasks. In the future, we will further investigate methods to narrow the gap between BiViTs and their full-precision counterparts.

Acknowledgement This work was supported by the National Key Research and Development Program of China (2022YFC3602601) and Key Research and Development Program of Zhejiang Province of China (2021C02037).

References

[1] Milad Alizadeh, Javier Fernández-Marqués, Nicholas D Lane, and Yarin Gal. A systematic study of binary neu-

ral networks' optimisation. In *International Conference on Learning Representations*, volume 63, page 81, 2019. 3

[2] Mehdi Bahri, Gaétan Bahl, and Stefanos Zafeiriou. Binary graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9492–9501, 2021. 6

[3] Haoli Bai, Wei Zhang, Lu Hou, Lifeng Shang, Jin Jin, Xin Jiang, Qun Liu, Michael R Lyu, and Irwin King. Binarybert: Pushing the limit of bert quantization. In *ACL/IJCNLP (1)*, 2021. 3, 5

[4] Yu Bai, Yu-Xiang Wang, and Edo Liberty. Proxquant: Quantized neural networks via proximal operators. In *International Conference on Learning Representations*, 2018. 1

[5] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013. 3

[6] Joseph Bethge, Christian Bartz, Haojin Yang, Ying Chen, and Christoph Meinel. Meliusnet: An improved network architecture for binary neural networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1439–1448, 2021. 3

[7] Adrian Bulat, Brais Martinez, and Georgios Tzimiropoulos. High-capacity expert binary networks. In *International Conference on Learning Representations*, 2020. 2, 3

[8] Adrian Bulat and Georgios Tzimiropoulos. Xnor-net++: Improved binary neural networks. *arXiv preprint arXiv:1909.13863*, 2019. 1

[9] Han Cai, Chuang Gan, and Song Han. Efficientvit: Enhanced linear attention for high-resolution low-computation visual recognition. *arXiv preprint arXiv:2205.14756*, 2022. 3

[10] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018. 9

[11] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 1

[12] Jieneng Chen, Yongyi Lu, Qihang Yu, Xiangde Luo, Ehsan Adeli, Yan Wang, Le Lu, Alan L Yuille, and Yuyin Zhou. Transunet: Transformers make strong encoders for medical image segmentation. *arXiv preprint arXiv:2102.04306*, 2021. 1

[13] Tianlong Chen, Yu Cheng, Zhe Gan, Lu Yuan, Lei Zhang, and Zhangyang Wang. Chasing sparsity in vision transformers: An end-to-end exploration. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 19974–19988. Curran Associates, Inc., 2021. 3

[14] Sharan Chetlur, Cliff Woolley, Philippe Vandermersch, Jonathan Cohen, John Tran, Bryan Catanzaro, and Evan Shelhamer. cudnn: Efficient primitives for deep learning. *arXiv preprint arXiv:1410.0759*, 2014. 8

[15] Matthieu Courbariaux and Yoshua Bengio. Binarynet: Training deep neural networks with weights and activations constrained to +1 or -1. *CoRR*, abs/1602.02830, 2016. 1

[16] Matthieu Courbariaux, Itay Hubara, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks: Training deep neural networks with weights and activations constrained to+ 1 or-1. *arXiv preprint arXiv:1602.02830*, 2016. 3

[17] Zihang Dai, Hanxiao Liu, Quoc V Le, and Mingxing Tan. Coatnet: Marrying convolution and attention for all data sizes. *Advances in Neural Information Processing Systems*, 34:3965–3977, 2021. 3

[18] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5, 6

[19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*, 2019. 1, 3

[20] Rui Ding, Haijun Liu, and Xichuan Zhou. Ie-net: Information-enhanced binary neural networks for accurate classification. *Electronics*, 11(6):937, 2022. 3

[21] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020. 1, 3

[22] Yuxin Fang, Bencheng Liao, Xinggang Wang, Jiemin Fang, Jiyang Qi, Rui Wu, Jianwei Niu, and Wenyu Liu. You only look at one sequence: Rethinking transformer in vision through object detection. *Advances in Neural Information Processing Systems*, 34:26183–26197, 2021. 1, 3

[23] Sahaj Garg, Anirudh Jain, Joe Lou, and Mitchell Nahmias. Confounding tradeoffs for neural network quantization. *arXiv preprint arXiv:2102.06366*, 2021. 2

[24] Benjamin Graham, Alaaeldin El-Nouby, Hugo Touvron, Pierre Stock, Armand Joulin, Hervé Jégou, and Matthijs Douze. Levit: a vision transformer in convnet's clothing for faster inference. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12259–12269, 2021. 1

[25] Jianyuan Guo, Kai Han, Han Wu, Yehui Tang, Xinghao Chen, Yunhe Wang, and Chang Xu. Cmt: Convolutional neural networks meet vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12175–12185, 2022. 3

[26] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 9

[27] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 7

[28] Yefei He, Luoming Zhang, Weijia Wu, and Hong Zhou. Binarizing by classification: Is soft function really necessary? *arXiv preprint arXiv:2205.07433*, 2022. 3

[29] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015. 6

[30] Lu Hou, Quanming Yao, and James Tin Yau Kwok. Loss-aware binarization of deep networks. In *5th International Conference on Learning Representations, ICLR 2017-Conference Track Proceedings*, page 000, 2017. 1

[31] Zejiang Hou and Sun-Yuan Kung. Multi-dimensional vision transformer compression via dependency guided gaussian process search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 3669–3678, June 2022. 3

[32] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 6

[33] Ding Jia, Kai Han, Yunhe Wang, Yehui Tang, Jianyuan Guo, Chao Zhang, and Dacheng Tao. Efficient vision transformers via fine-grained manifold distillation. *arXiv preprint arXiv:2107.01378*, 2021. 1

[34] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[35] Andrew Lavin and Scott Gray. Fast algorithms for convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4013–4021, 2016. 8

[36] Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Ng. Efficient sparse coding algorithms. *Advances in neural information processing systems*, 19, 2006. 4

[37] Ang Li and Simon Su. Accelerating binarized neural networks via bit-tensor-cores in turing gpus. *IEEE Transactions on Parallel and Distributed Systems*, 32(7):1878–1891, 2021. 8

[38] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. *arXiv preprint arXiv:2203.16527*, 2022. 1

[39] Yanjing Li, Sheng Xu, Baochang Zhang, Xianbin Cao, Peng Gao, and Guodong Guo. Q-vit: Accurate and fully quantized low-bit vision transformer. *arXiv preprint arXiv:2210.06707*, 2022. 3

[40] Yanyu Li, Geng Yuan, Yang Wen, Eric Hu, Georgios Evangelidis, Sergey Tulyakov, Yanzhi Wang, and Jian Ren. Efficientformer: Vision transformers at mobilenet speed. *arXiv preprint arXiv:2206.01191*, 2022. 3

[41] Yawei Li, Kai Zhang, Jiezhang Cao, Radu Timofte, and Luc Van Gool. Localvit: Bringing locality to vision transformers. *arXiv preprint arXiv:2104.05707*, 2021. 3

[42] Zhikai Li and Qingyi Gu. I-vit: integer-only quantization for efficient vision transformer inference. *arXiv preprint arXiv:2207.01405*, 2022. 1

[43] Zhexin Li, Tong Yang, Peisong Wang, and Jian Cheng. Q-vit: Fully differentiable quantization for vision transformer. *arXiv preprint arXiv:2201.07703*, 2022. 1, 3

[44] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 9

[45] Yang Lin, Tianyu Zhang, Peiqin Sun, Zheng Li, and Shuchang Zhou. Fq-vit: Post-training quantization for fully quantized vision transformer. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 1173–1179, 2022. 1, 3

[46] Chunlei Liu, Wenrui Ding, Xin Xia, Baochang Zhang, Jiaxin Gu, Jianzhuang Liu, Rongrong Ji, and David Doermann. Circulant binary convolutional networks: Enhancing the performance of 1-bit dcnns with circulant back propagation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2691–2699, 2019. 3

[47] Jing Liu, Zizheng Pan, Haoyu He, Jianfei Cai, and Bohan Zhuang. Ecoformer: Energy-saving attention with linear complexity. In *NeurIPS*, 2022. 2

[48] Xingyu Liu, Jeff Pool, Song Han, and William J Dally. Efficient sparse-winograd convolutional neural networks. *arXiv preprint arXiv:1802.06367*, 2018. 3

[49] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 1, 3, 5, 6, 9

[50] Zechun Liu, Barlas Oguz, Aasish Pappu, Lin Xiao, Scott Yih, Meng Li, Raghuraman Krishnamoorthi, and Yashar Mehdad. Bit: Robustly binarized multi-distilled transformer. *arXiv preprint arXiv:2205.13016*, 2022. 1, 2, 3, 4, 5, 6

[51] Zechun Liu, Zhiqiang Shen, Shichao Li, Koen Helwegen, Dong Huang, and Kwang-Ting Cheng. How do adam and training strategies help bnns optimization. In *International Conference on Machine Learning*, pages 6936–6946. PMLR, 2021. 3

[52] Zechun Liu, Zhiqiang Shen, Marios Savvides, and Kwang-Ting Cheng. Reactnet: Towards precise binary neural network with generalized activation functions. In *European conference on computer vision*, pages 143–159. Springer, 2020. 1, 2, 3, 6

[53] Zhenhua Liu, Yunhe Wang, Kai Han, Wei Zhang, Siwei Ma, and Wen Gao. Post-training quantization for vision transformer. *Advances in Neural Information Processing Systems*, 34:28092–28103, 2021. 3

[54] Zechun Liu, Baoyuan Wu, Wenhan Luo, Xin Yang, Wei Liu, and Kwang-Ting Cheng. 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)*, pages 722–737, 2018. 1, 3

[55] Brais Martinez, Jing Yang, Adrian Bulat, and Georgios Tzimiropoulos. Training binary neural networks with real-to-binary convolutions. In *International Conference on Learning Representations*, 2019. 3, 6, 8

[56] Asit Mishra and Debbie Marr. Apprentice: Using knowledge distillation techniques to improve low-precision network accuracy. In *International Conference on Learning Representations*, 2018. 3

[57] Asit Mishra, Eriko Nurvitadhi, Jeffrey J Cook, and Debbie Marr. Wrpn: Wide reduced-precision networks. In *International Conference on Learning Representations*, 2018. 3

[58] Bowen Pan, Rameswar Panda, Yifan Jiang, Zhangyang Wang, Rogerio Feris, and Aude Oliva. Ia-red: Interpretability-aware redundancy reduction for vision transformers. *Advances in Neural Information Processing Systems*, 34:24898–24911, 2021. 1

[59] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 6

[60] Haotong Qin, Yifu Ding, Mingyuan Zhang, YAN Qinghua, Aishan Liu, Qingqing Dang, Ziwei Liu, and Xianglong Liu. Bibert: Accurate fully binarized bert. In *International Conference on Learning Representations*, 2021. 1, 2, 3, 4, 5, 6

[61] Haotong Qin, Ruihao Gong, Xianglong Liu, Mingzhu Shen, Ziran Wei, Fengwei Yu, and Jingkuan Song. Forward and backward information retention for accurate binary neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2250–2259, 2020. 1, 2, 3

[62] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *European conference on computer vision*, pages 525–542. Springer, 2016. 1, 3, 6

[63] Zhuoran Shen, Mingyuan Zhang, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Efficient attention: Attention with linear complexities. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 3531–3539, 2021. 3

[64] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7262–7272, 2021. 1

[65] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021. 1

[66] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, volume 139, pages 10347–10357, July 2021. 3, 6

[67] Hugo Touvron, Matthieu Cord, and Herve Jegou. Deit iii: Revenge of the vit. *arXiv preprint arXiv:2204.07118*, 2022. 1

[68] Zhijun Tu, Xinghao Chen, Pengju Ren, and Yunhe Wang. Adabin: Improving binary neural networks with adaptive binary sets. In *European Conference on Computer Vision*, pages 379–395. Springer, 2022. 2

[69] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2, 3

[70] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 568–578, 2021. 3

[71] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvt v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3):415–424, 2022. 3

[72] Ross Wightman. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019. 6

[73] Jiayu Wu, Qixiang Zhang, and Guoxi Xu. Tiny imagenet challenge. *Technical report*, 2017. 6

[74] Huanrui Yang, Hongxu Yin, Pavlo Molchanov, Hai Li, and Jan Kautz. Nvit: Vision transformer compression and parameter redistribution. *arXiv preprint arXiv:2110.04869*, 2021. 3

[75] Hao Yu and Jianxin Wu. A unified pruning framework for vision transformers. *arXiv preprint arXiv:2111.15127*, 2021. 1

[76] Zhihang Yuan, Chenhao Xue, Yiqi Chen, Qiang Wu, and Guangyu Sun. Ptq4vit: Post-training quantization framework for vision transformers. *arXiv preprint arXiv:2111.12293*, 2021. 3

[77] Yanhong Zeng, Huan Yang, Hongyang Chao, Jianbo Wang, and Jianlong Fu. Improving visual quality of image synthesis by a token-based generator with transformers. *Advances in Neural Information Processing Systems*, 34:21125–21137, 2021. 3

[78] Dongqing Zhang, Jiaolong Yang, Dongqiangzi Ye, and Gang Hua. Lq-nets: Learned quantization for highly accurate and compact deep neural networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 365–382, 2018. 4

[79] Luoming Zhang, Yefei He, Zhenyu Lou, Xin Ye, Yuxing Wang, and Hong Zhou. Root quantization: a self-adaptive supplement ste. *Applied Intelligence*, pages 1–10, 2022. 3

[80] Wenqiang Zhang, Zilong Huang, Guozhong Luo, Tao Chen, Xinggang Wang, Wenyu Liu, Gang Yu, and Chunhua Shen. Topformer: Token pyramid transformer for mobile semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12083–12093, 2022. 3

[81] Zizhao Zhang, Han Zhang, Long Zhao, Ting Chen, Sercan Ö Arik, and Tomas Pfister. Nested hierarchical transformer: Towards accurate, data-efficient and interpretable visual understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 3417–3425, 2022. 3, 4, 5, 6

[82] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6881–6890, 2021. 1

[83] Shuchang Zhou, Yuxin Wu, Zekun Ni, Xinyu Zhou, He Wen, and Yuheng Zou. Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients. *arXiv preprint arXiv:1606.06160*, 2016. 1

[84] Mingjian Zhu, Yehui Tang, and Kai Han. Vision transformer pruning. *arXiv preprint arXiv:2104.08500*, 2021. 1

[85] Shilin Zhu, Xin Dong, and Hao Su. Binary ensemble neural network: More bits per network or more networks per bit? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4923–4932, 2019. 3

[86] Bohan Zhuang, Chunhua Shen, Mingkui Tan, Peng Chen, Lingqiao Liu, and Ian Reid. Structured binary neural networks for image recognition. *International Journal of Computer Vision*, 130(9):2081–2102, 2022. 1, 2

[87] Bohan Zhuang, Chunhua Shen, Mingkui Tan, Lingqiao Liu, and Ian Reid. Towards effective low-bitwidth convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7920–7928, 2018. 6